Contents

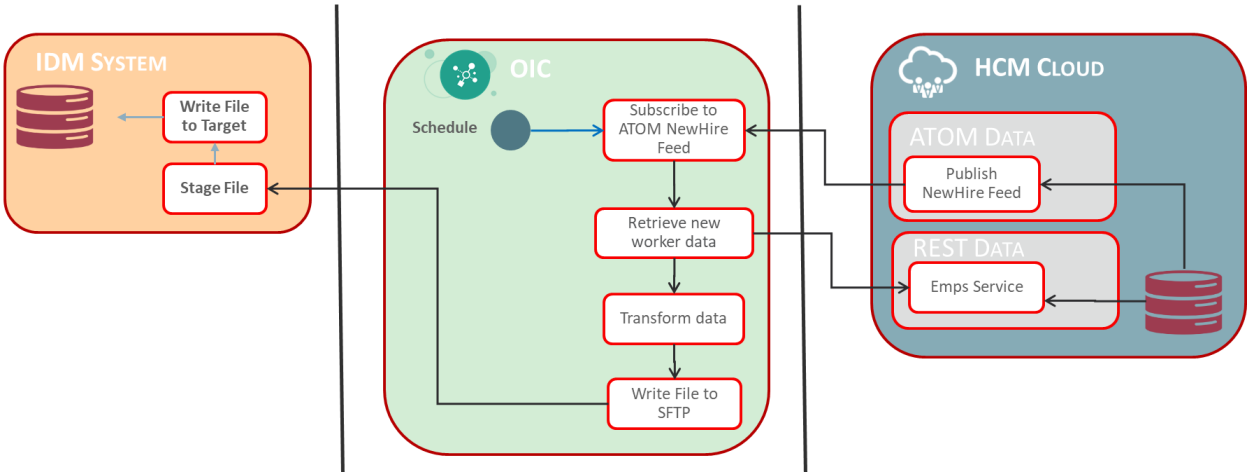
EXECUTIVE SUMMARY 1
USECASE BACKGROUND 1
Goal 2
OBJECTIVE 2
BUILDING THE Directory Synchronization USE CASE 2
Prerequisites 2
Creating Connections 2
Creating the Directory Synchronization Integration 2
Schedule Parameters 4
Function Call (Action) 4
getNewHireATOMFeed (HCM Cloud Adapter) 11
Map to getNewHireATOMFeed 14
countOfNewHires 15
Switch Action 16
Add Stop Action 19
StageFileRef (Assign Activity) 20
ForEachEntry 21
getEmployeeDetails (REST Adapter) 23
Map to getEmployeeDetails 26
writeNewUserRecord to Staging 28
Map to WriteRecordToStage 32
WriteStageFileToFTP (FTP Adapter) 34
Map to WriteStageFileToFTP 35
assignATOMLRDT 36
Tracking 36
Activate and Run the Integration 36
Validating Extracted File presence on FTP Server 38

EXECUTIVE SUMMARY

Oracle Integration provides native connectivity to Oracle and non-Oracle Software as a Service (SaaS) and On-premises applications, such as Oracle ERP Cloud, Oracle Service Cloud, HCM Cloud, Salesforce.com, Workday, EBS, SAP, NetSuite and so on. Oracle Integration adapters simplify connectivity by handling the underlying complexities of connecting to applications using industry-wide best practices. You only need to create a connection that provides minimal connectivity information for each system.

USECASE BACKGROUND

This use case explores the use of OIC with HCM REST and ATOM services in order to satisfy the downstream needs of an Identity Management System hosted by a customer.
The following diagram illustrates the proposed interaction between the systems involved in this use case.



Goal

This document walks you through the steps needed to replicate in your own environment what has been released in the demo title: *Directory Synchronization*

OBJECTIVE

This document walks you through the steps needed to replicate this use case in your environment

BUILDING THE Directory Synchronization USE CASE

This section works through the steps that are required to build the integration from scratch.

Prerequisites

You will need access to the following applications and artifacts:

- Oracle Integration (OIC)
- HCM Cloud R13+

Note: For ClassID & StudentId: Please check with the instructor.

Creating Connections

The following Connections have been created and configured. You will be using these connections for creating Integration flows

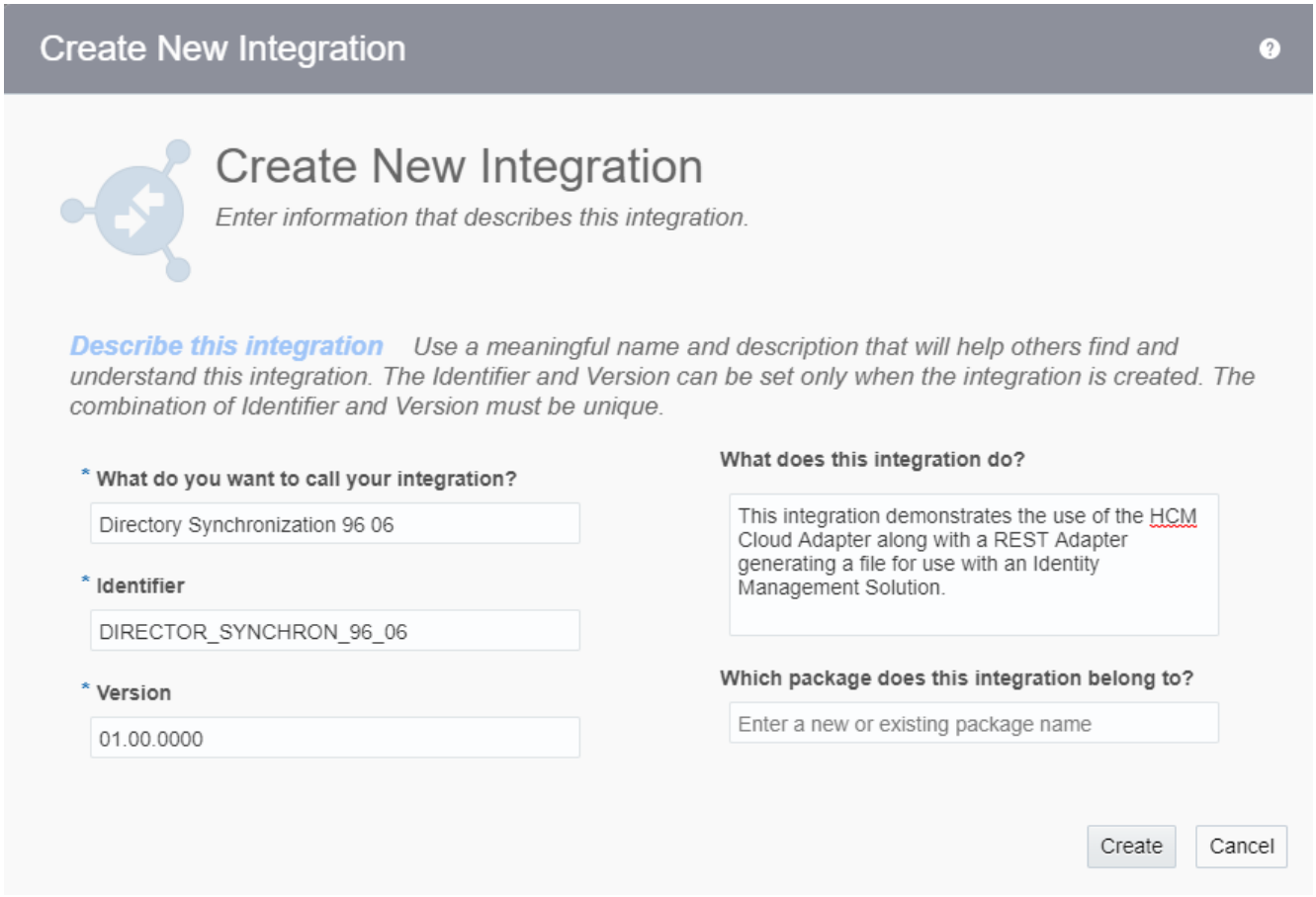
Connection Name	Connection Type
HCM Conn 96 06	HCM Adapter
HCM REST Conn 96 06	REST Adapter
FTP Conn 96 06	FTP Adapter

Creating the Directory Synchronization Integration

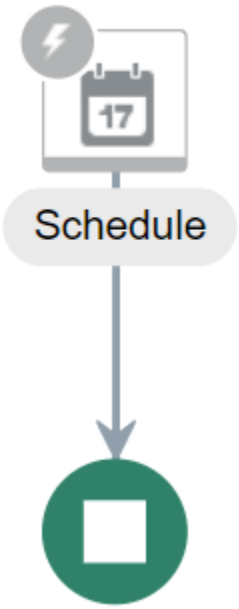
1. On the Oracle Integration home page, click **Integrations**.
2. On the Integrations page, click **Create**. The Create Integration - Select a Style/Pattern dialog is displayed.

3. Select **Scheduled Orchestration** type of integration. The **Create New** Integration dialog is displayed.
4. Enter the following information:

Field Element	Value
What do you want to call your integration?	Directory Synchronization <ClassId> <StudentId> Ex: Directory Synchronization 96 06
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of 01.00.0000. Or, if you want to change the version number, enter the version using numbers only in this format: xx.xx.xxxx.
What does this integration do?	This integration demonstrates the use of the HCM Cloud Adapter along with a REST Adapter generating a file for use with an Identity Management Solution.
Which package does this integration belong to?	*Leave blank*



5. Click **Create**. The integration canvas is displayed.



Schedule Parameters

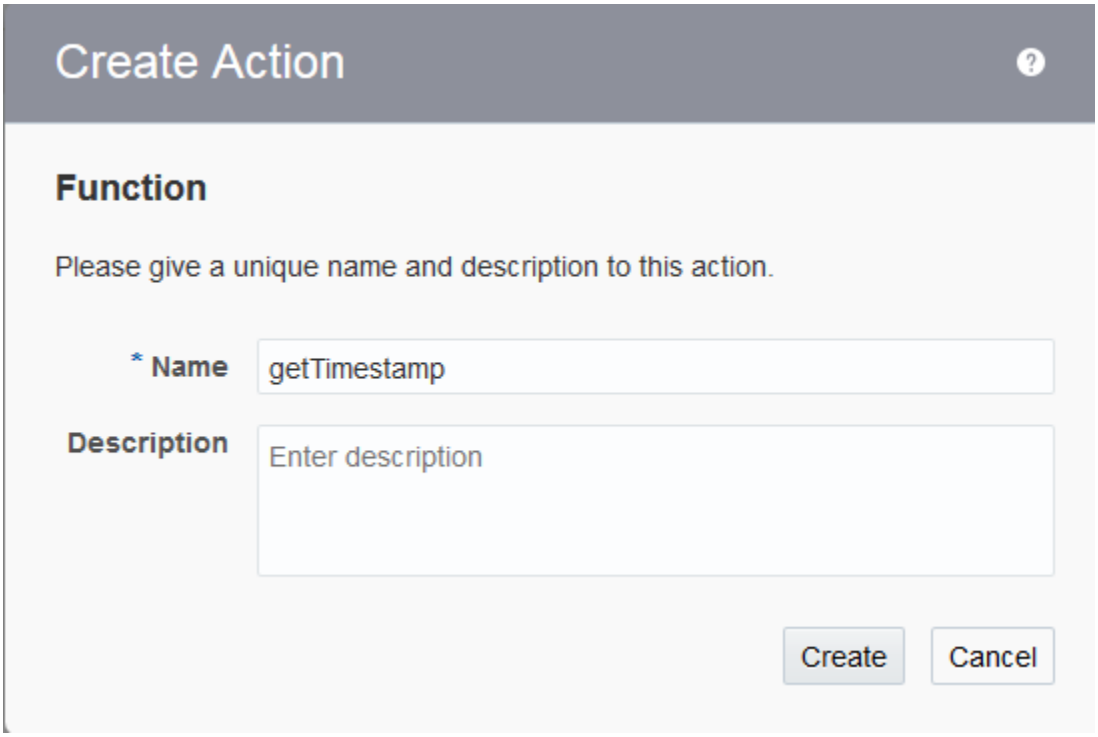
The first step in defining the integration is to retrieve from **Oracle Integration** the last time the integration polled the ATOM service. This is known as the Last Run Date and Time. This allows you to avoid pulling duplicate records from the ATOM feed.

1. Click the **Schedule** icon, then select **Edit**. The **Schedule Parameters** configuration page will be displayed.
2. On the empty **Parameter Name** row, click the **+** icon.
3. Enter ATOMLastRunDateTime as the **parameter name**,
4. Enter Last successful processed ATOM pull as the optional **Description**.
5. Enter 2019-01-01T00:00:00.000Z in the **Value** column. This is to set the initial period of time to retrieve the ATOM feed from HCM Cloud to the 1st of January 2019. You may need to adjust this to a closer DateTime value depending on when you are building this integration. After the first time the integration is run, this value will be updated to reflect the actual Date and Time values.
6. Click **Close**. This will return you to the integration canvas.
7. Click on Save to save your integration flow.

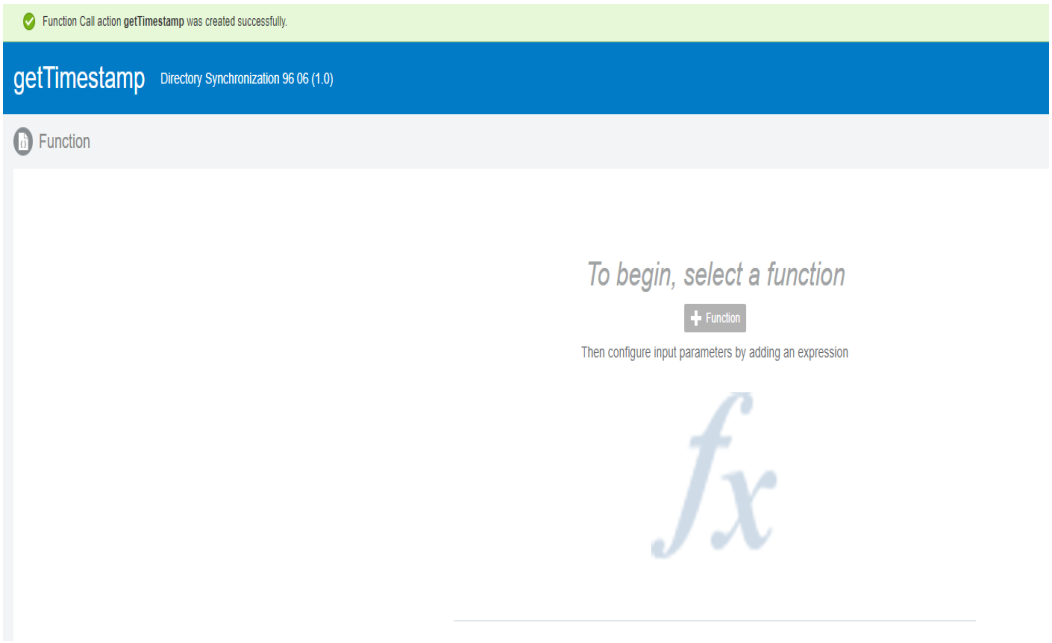
Function Call (Action)

The next step within the integration is used to set the current Date and Time of when the current ATOM poll is occurring. This will then be used as the ATOMLastRunDateTime variable the next time the integration is scheduled.

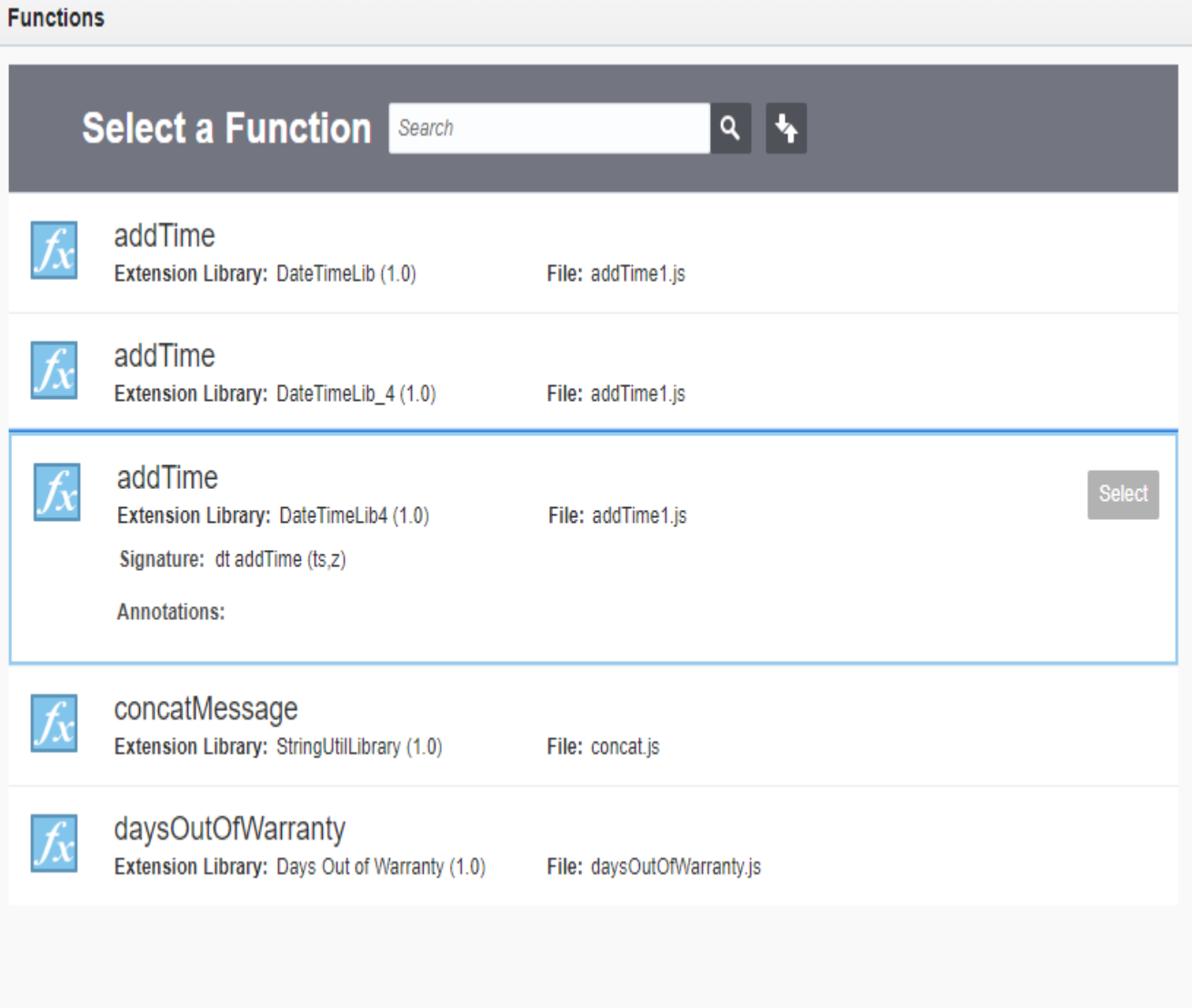
1. In the right navigation pane, click **Actions**
2. Drag the **Function Call** icon to the **+** sign following the **Schedule** label.
3. Enter getTimestamp for the function call action when prompted and Click **Create**. This will open the **Function Call** configuration dialogue



4. Click the **+Function** button.

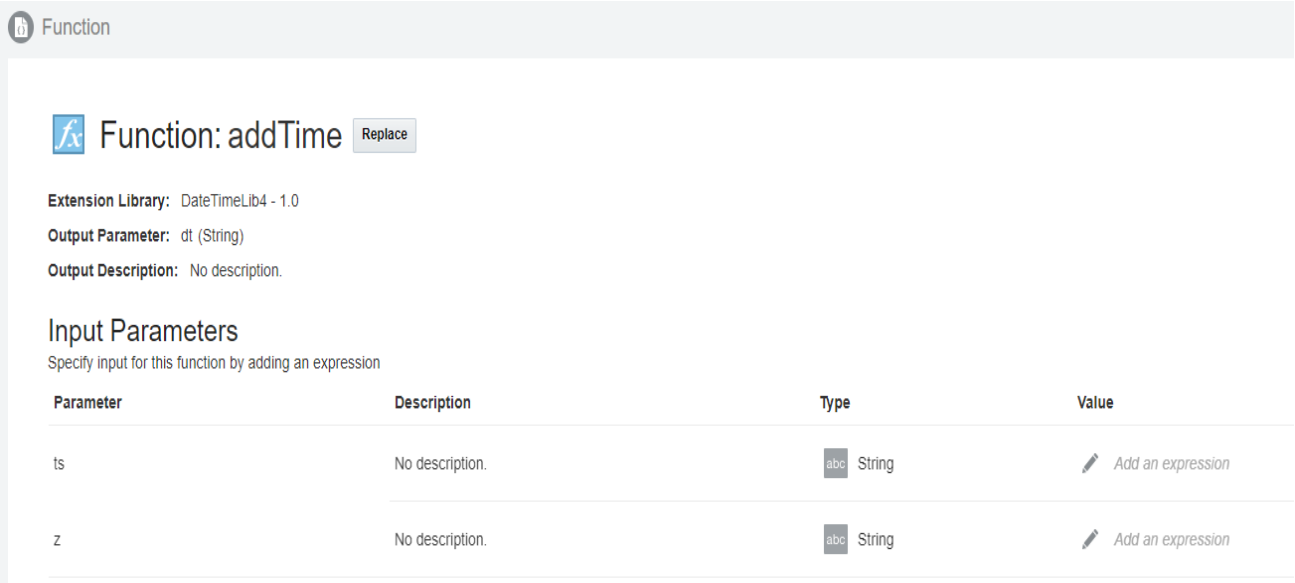


5. The **Select a Function** dialogue window appears.

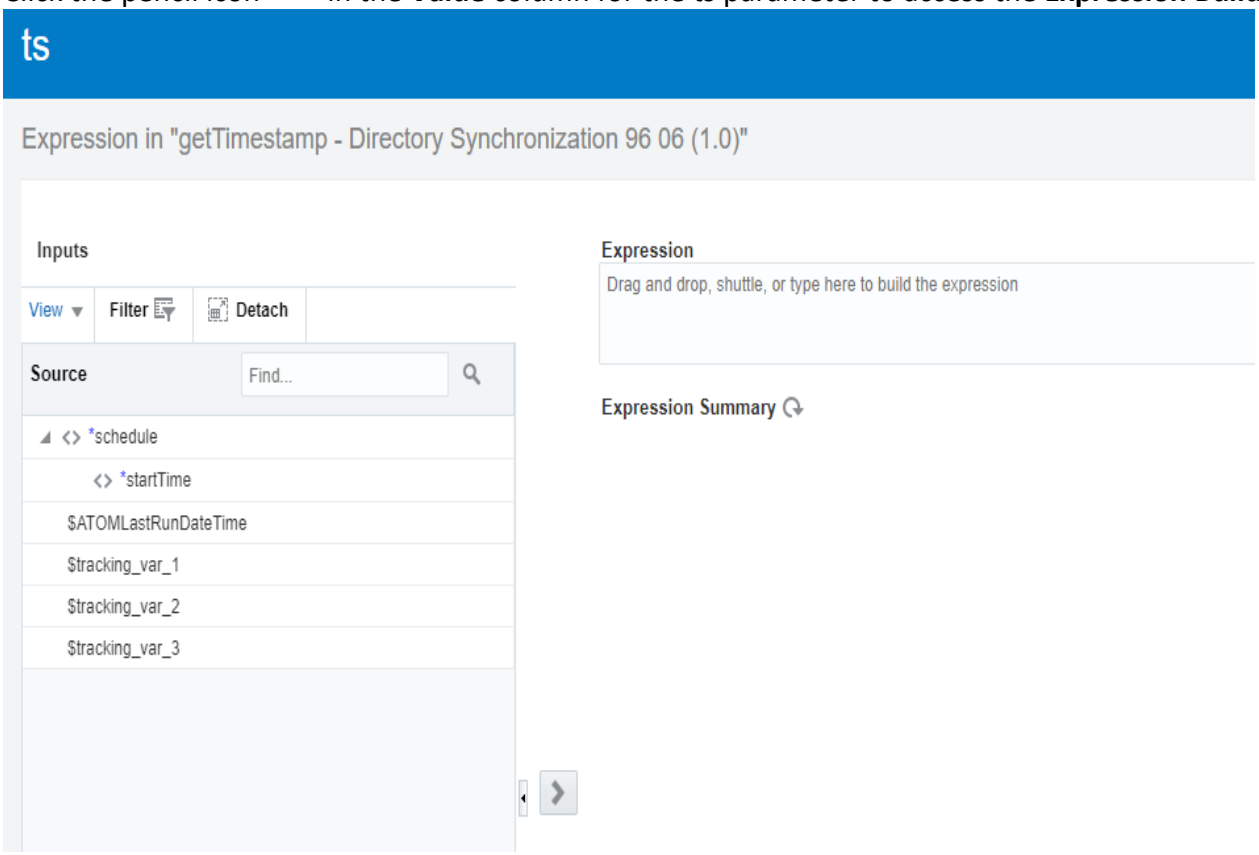


*Note: depending on your OIC instance, the available libraries available will differ from those shown above. If there isn't **addTime** function Register the **DateTimeLib4_1.0.jar** navigating to the library section of OIC.*

6. Find and **Click** addTime (Refer above screenshot)
7. Click the **Select** button in the function's row. The configuration page for the addTime function is displayed. It shows the details of the selected function including the input and output parameters.



8. Click the pencil icon  in the **Value** column for the ts parameter to access the **Expression Builder**.



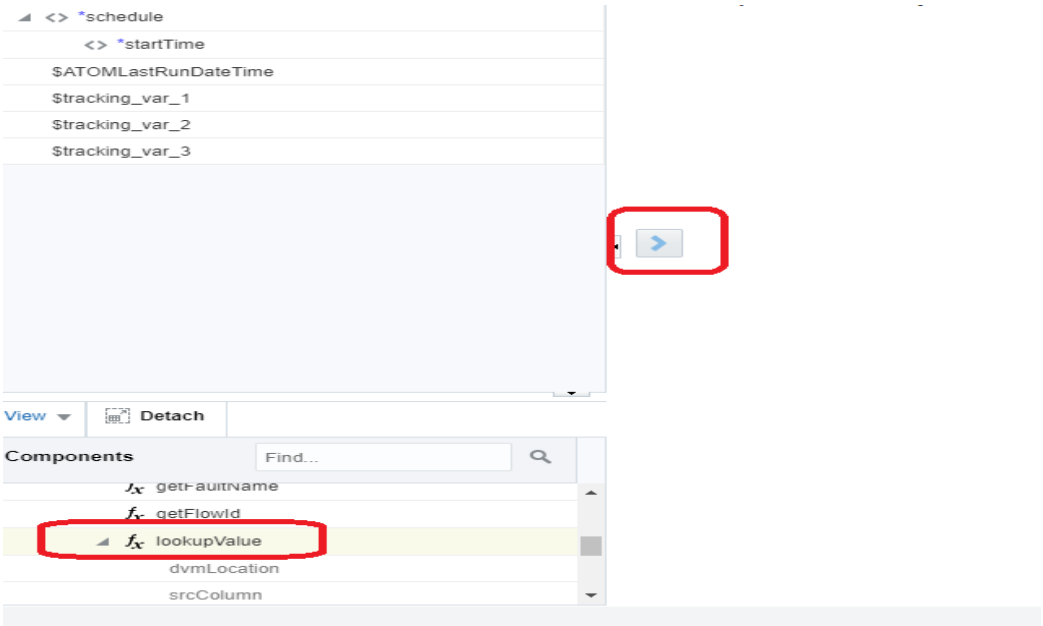
9. Enter the following text directly into the **Expression** text box

   concat(substring-before(/nssrcmpr:schedule/nssrcmpr:startTime,"."),".000Z")

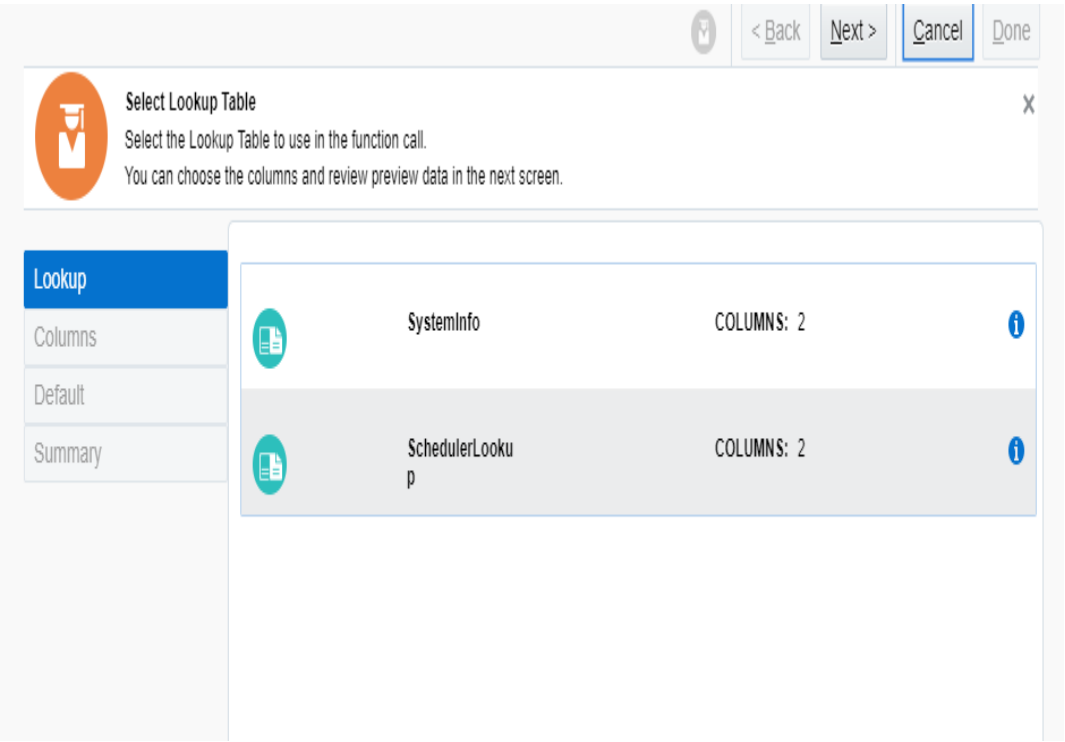
10. Click **Validate** in the title bar to validate the parameter.
11. Click **Close**

You will be returned to the **Function Call** page.

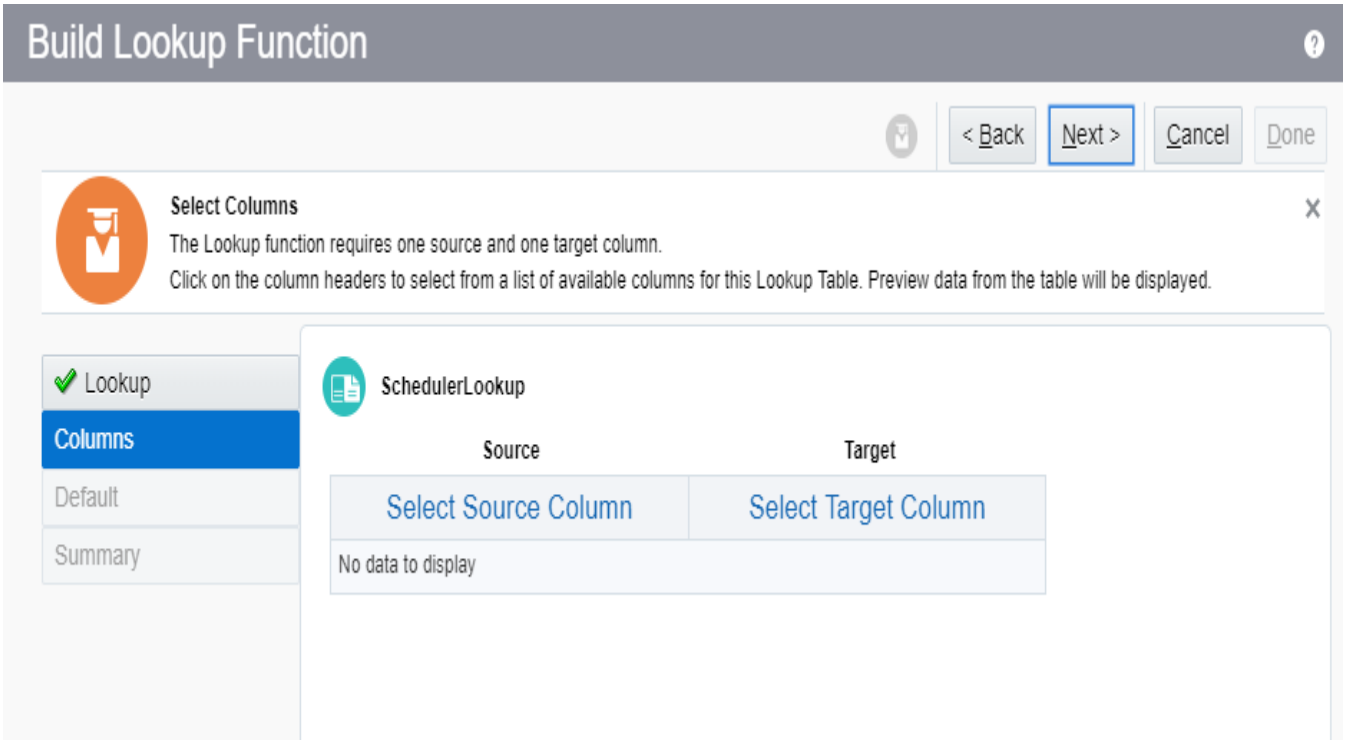
12. Click the pencil icon in the **Value** column for the z parameter to access the **Expression Builder**.
13. Select the lookupValue element which is under **Components>Functions>Integration Cloud** and click on Move button.



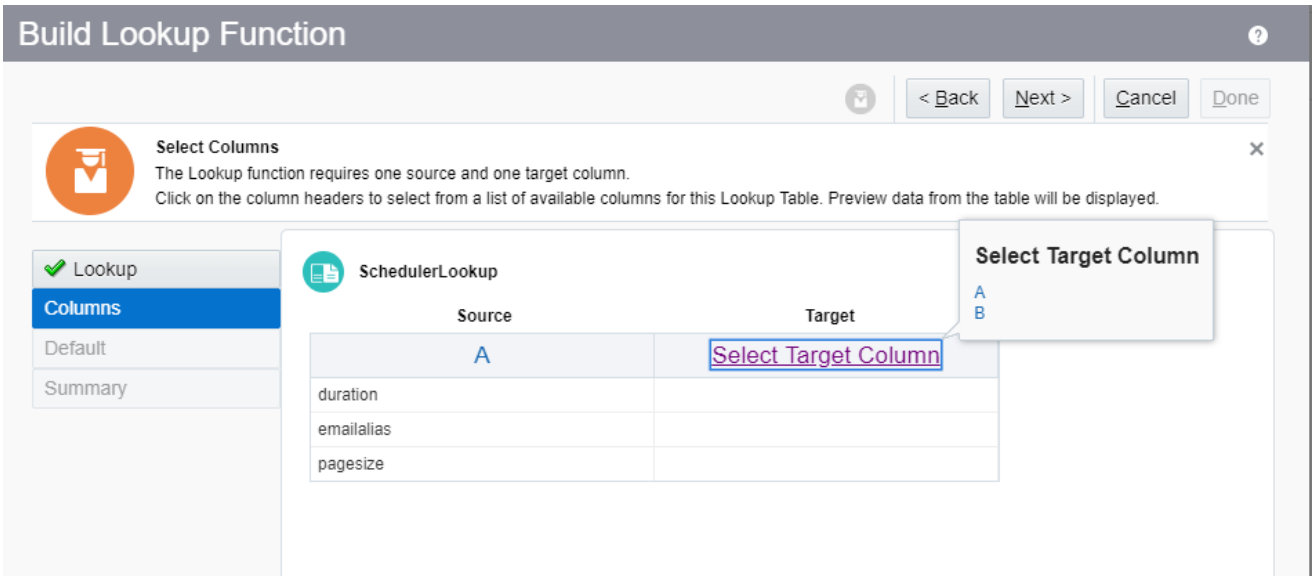
The **Build Lookup Function** configuration page is displayed.



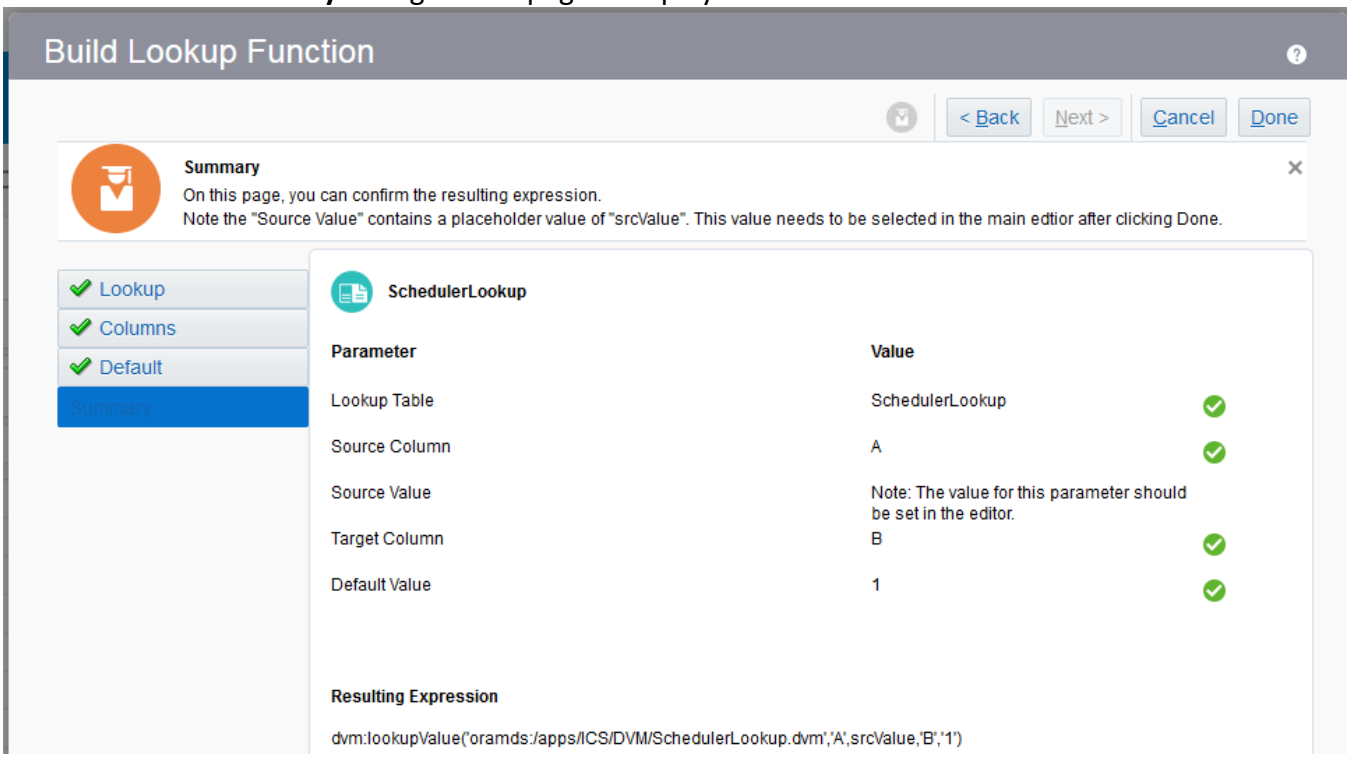
14. Find and select the **SchedulerLookup** Lookup Table.
15. Click **Next**. The **Select Columns** configuration page is displayed.



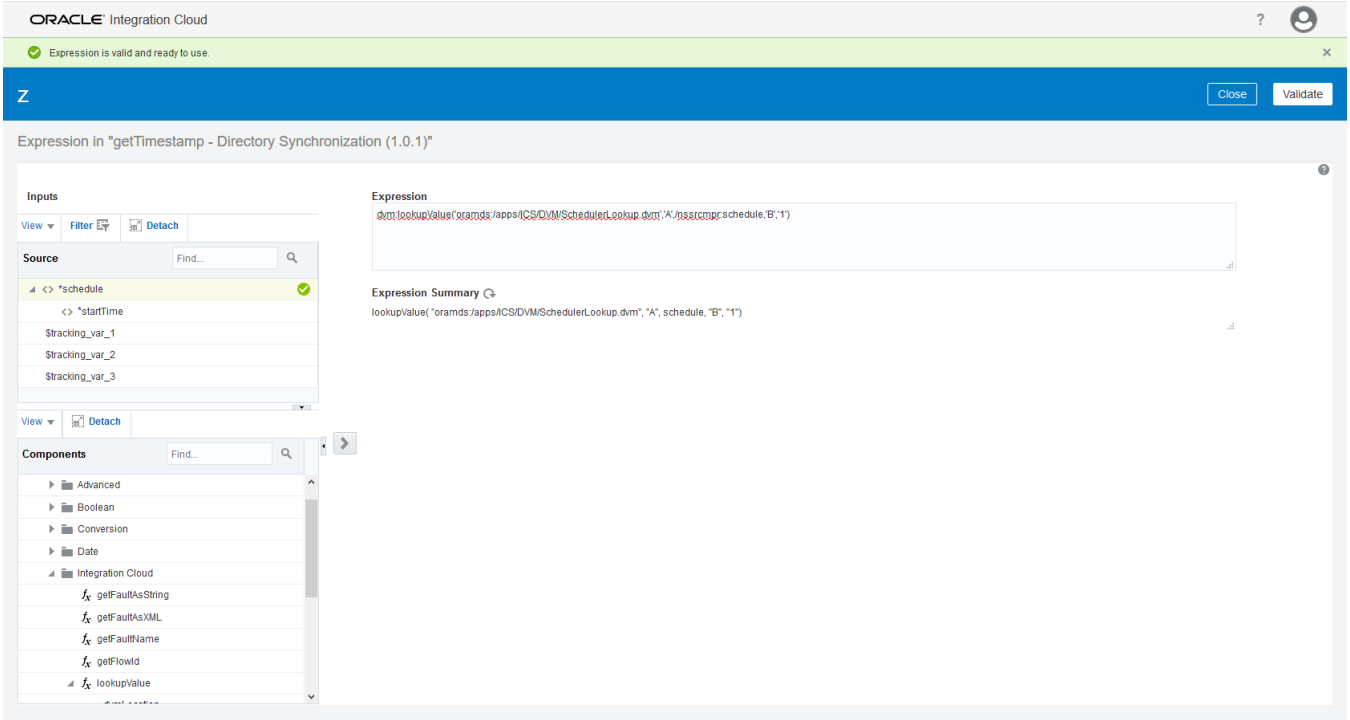
16. Click on Select Source Column, select A and click on Select Target Column, select B



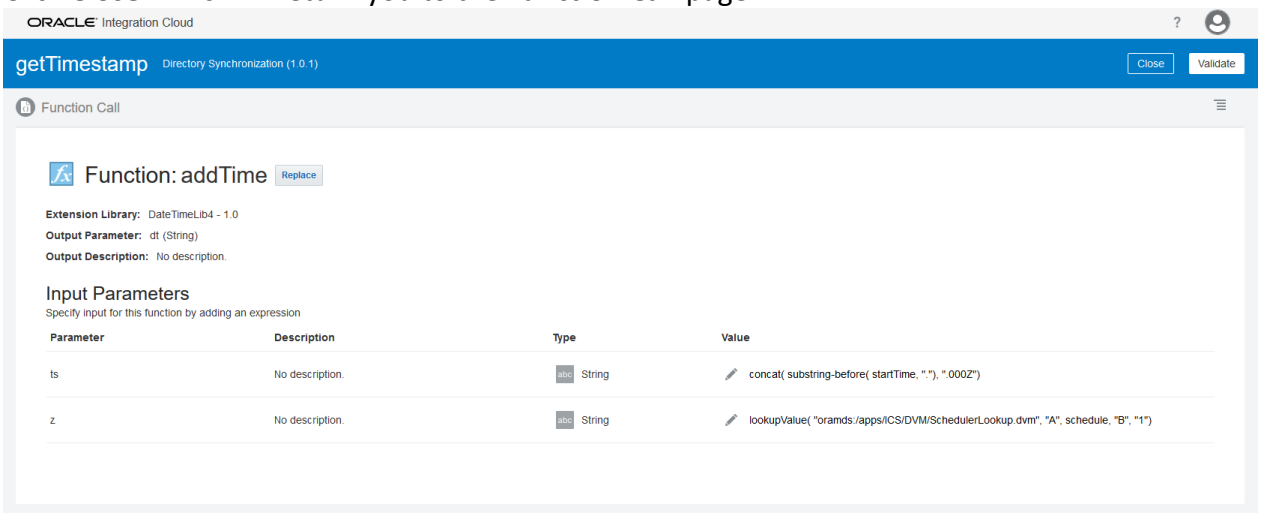
17. Click **Next**. The **Default configuration** page is displayed, **Default value** 1
18. Click **Next**. The **Summary** configuration page is displayed.



19. Click **Done**. This will return you to the **Expression Builder** page.



20. In the **Expression** text box, replace the SrcValue with the “schedule” parameter by dragging from the Source menu.
21. Click **Validate** in the title bar to validate the parameter.
22. Click **Close.** This will return you to the **Function Call** page.



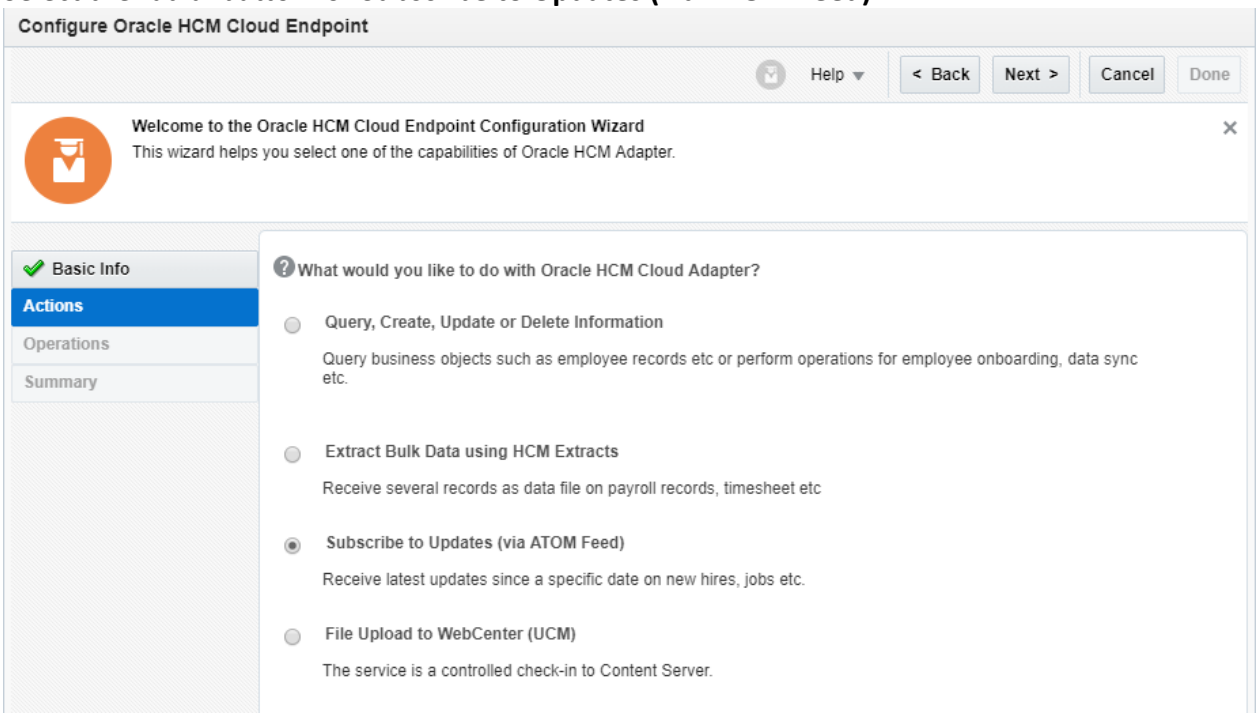
23. Click **Validate** in the title bar to validate the **Function Call**.
24. Click **Close.** This will return you to the integration canvas.

getNewHireATOMFeed (HCM Cloud Adapter)

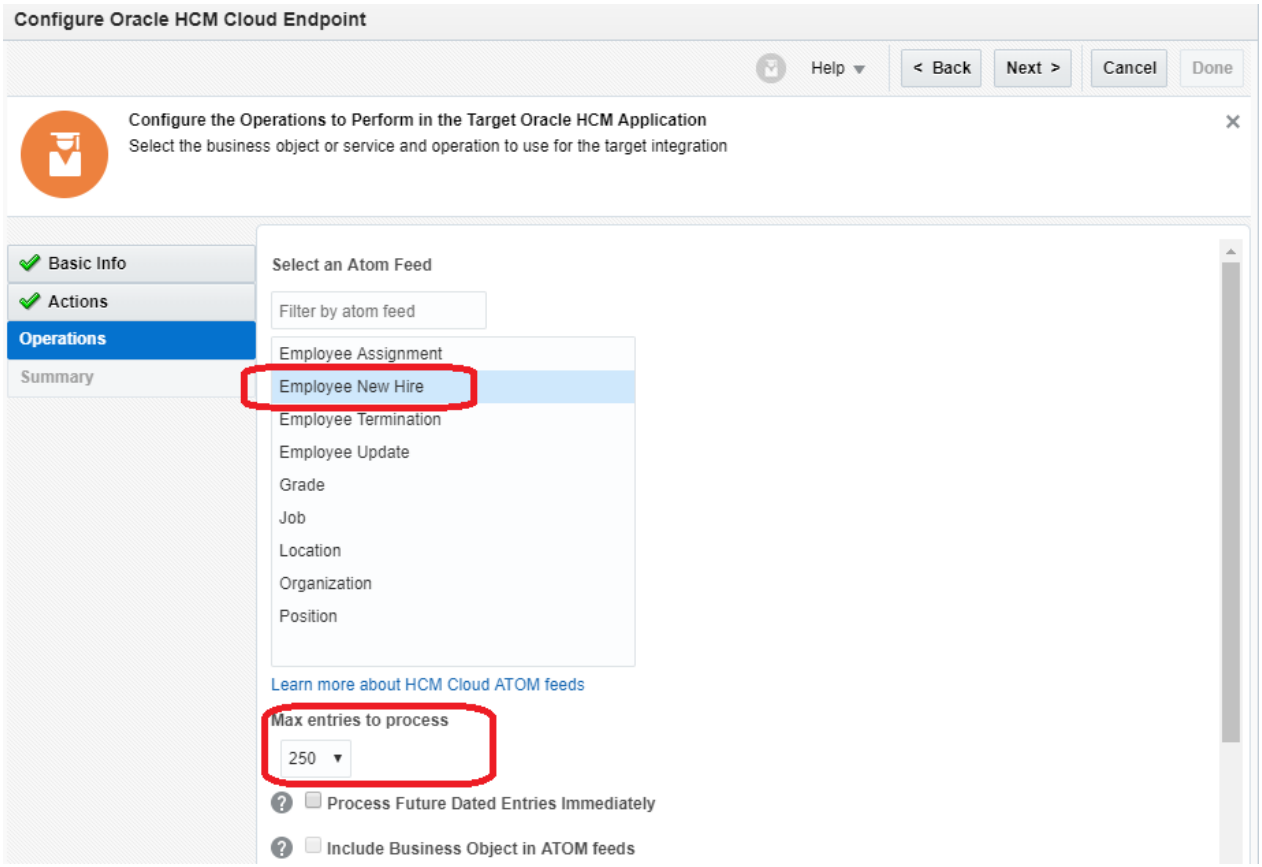
The next task in the integration is the retrieval of the HCM Cloud ATOM feed accessed via **the HCM Cloud Adapter**.

1. In the right navigation pane, click **Invokes**.

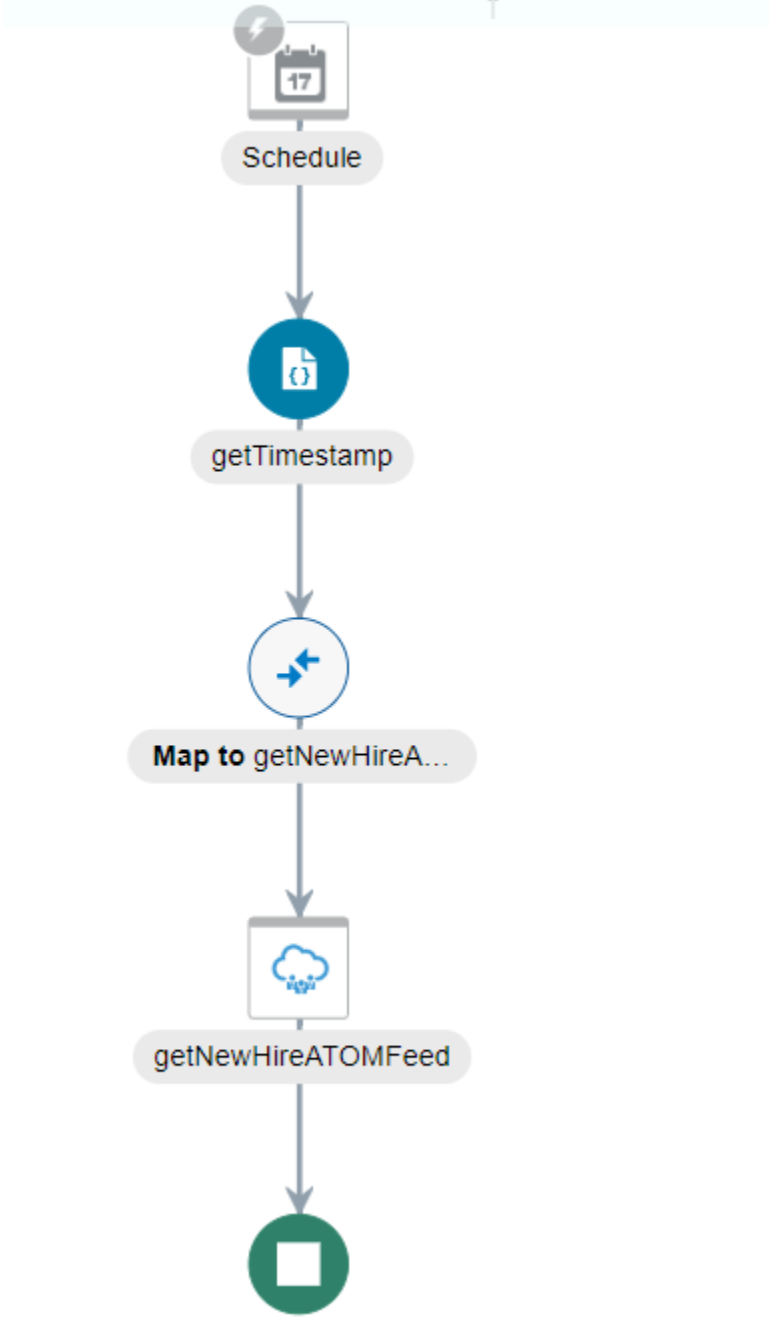
2. Click **Oracle HCM Cloud**, then drag the HCM Conn 96 06 to the **+** sign following the GetTimestamp action. The **Oracle HCM Cloud Adapter** configuration wizard is displayed.
3. Enter getNewHireATOMFeed in the **What do you want to call your endpoint?**
4. Click **Next**. This will display **Actions** tab.
5. Select the radial button for **Subscribe to Updates (via ATOM Feed)**.



6. Click **Next**. This will display the **Operations** tab.
7. Select the Employee New Hire as the **Select an ATOM Feed** value.
8. Select in the drop down **Max entries to process** to 250. This will ensure that most customers have all of their new hires returned in the ATOM Feed response.



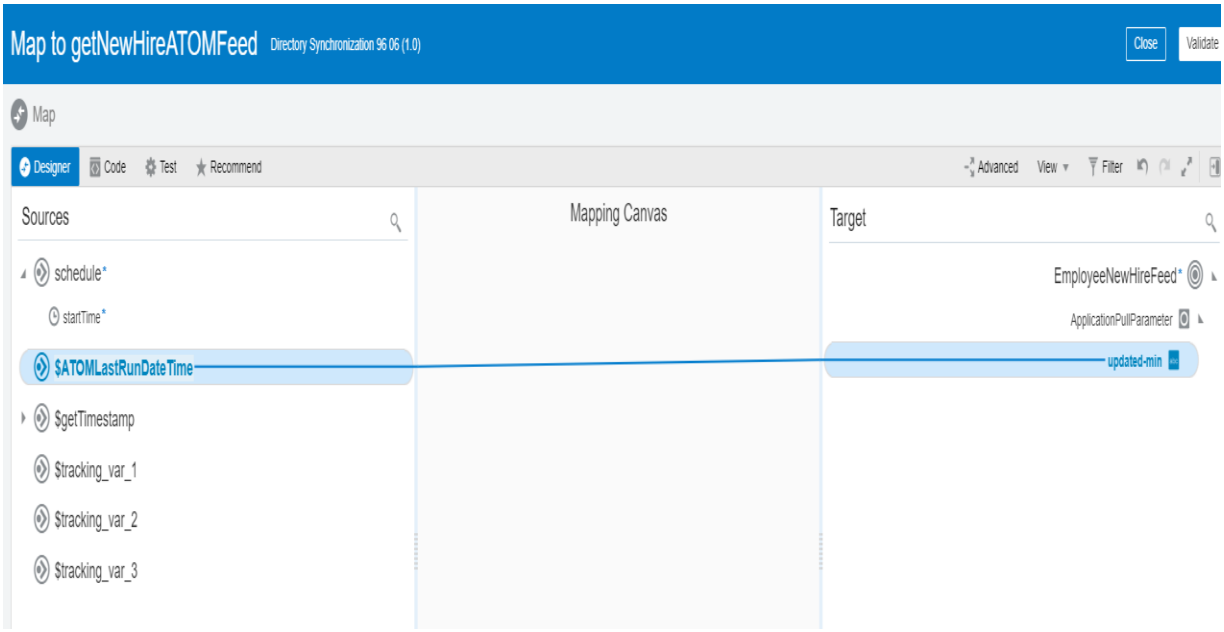
9. Click **Next**. This will display the **Summary** tab.
10. Click **Done**, this will return you to the integration canvas.
11. Click on Save



Map to getNewHireATOMFeed

A Map to icon will now be displayed between the **getTimestamp** and **getNewHireATOMFeed**. This represents the input values parameters for the ATOM Feed just added.

1. Click the Mapper icon (**Map to getNewHireATOMFeed**).
2. Click **Edit** to invoke the mapper.
3. Select the **$ATOMLastRunDateTime** from the **Sources** section and drag it on to the **updated-min Target** field.

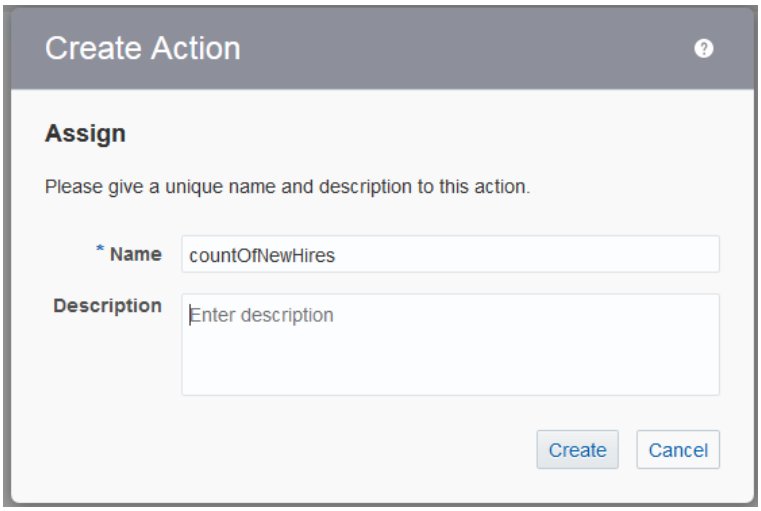


4. Click Validate and Click on **Close.** This will return you to the integration canvas.
5. Click on Save

countOfNewHires

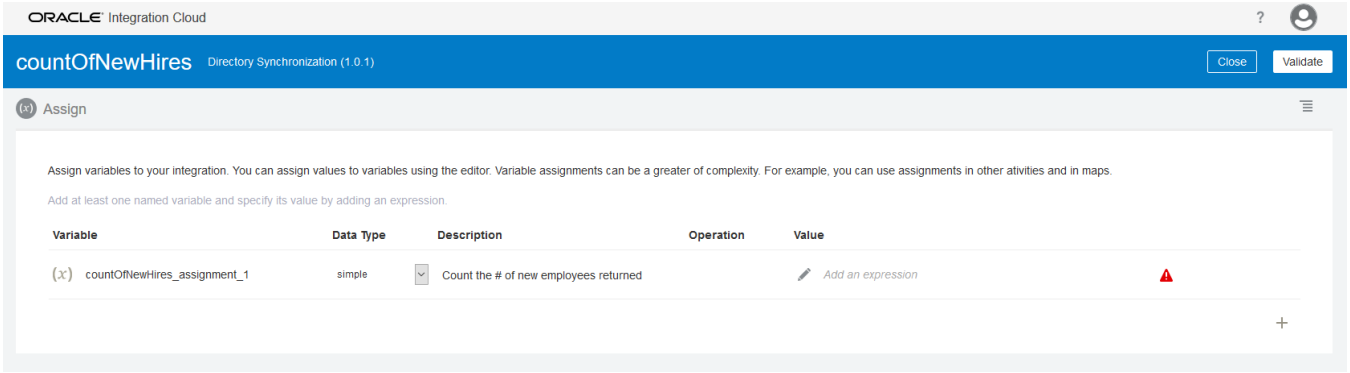
The next task in the integration is to count the number of new hires returned in the ATOM feed. By default the ATOM feed does not return the total number of new hires, so OIC needs to be able to derive this. This is achieved by adding an **Assign Action** to the integration.

1. In the right navigation pane, click **Actions**
2. Drag the **Assign** icon to the **+** sign following the **getNewHireATOMFeed** label.
3. Enter "countOfNewHires" for the Assign action when prompted,
   Click **Create**. This will open the **Expression** page configuration

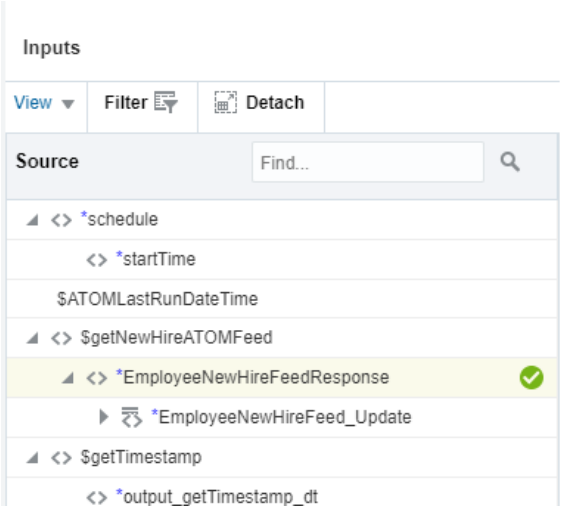


8. Click the **+** icon.

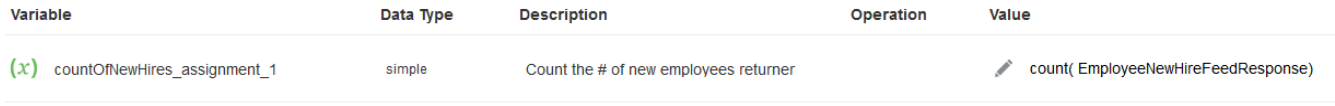
9. Enter countOfNewHires_assignment_1 as the **variable name**, Count the # of new employees returned as the optional **Description**



10. Select the pencil to edit the **Value**. This will open the **Expression** builder.
11. Select and drag the **Count** Function (Available under Node-set) into the **Expression** text box.
12. Select the **EmployeeNewFeedResponse** value from the **Source** component and drag into **Count** expression within the **Expression** text box.



13. Click **Validate** in the title bar to validate the Expression.
14. Click **Close.** This will return you to the **Parameter** configuration page.



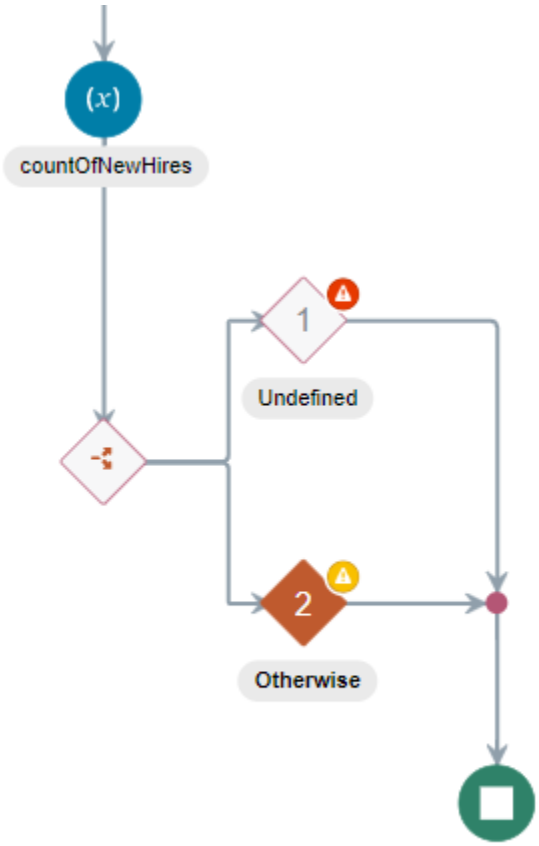
15. Click Validate and Click on **Close.** This will return you to the integration canvas.
16. Click on Save

Switch Action

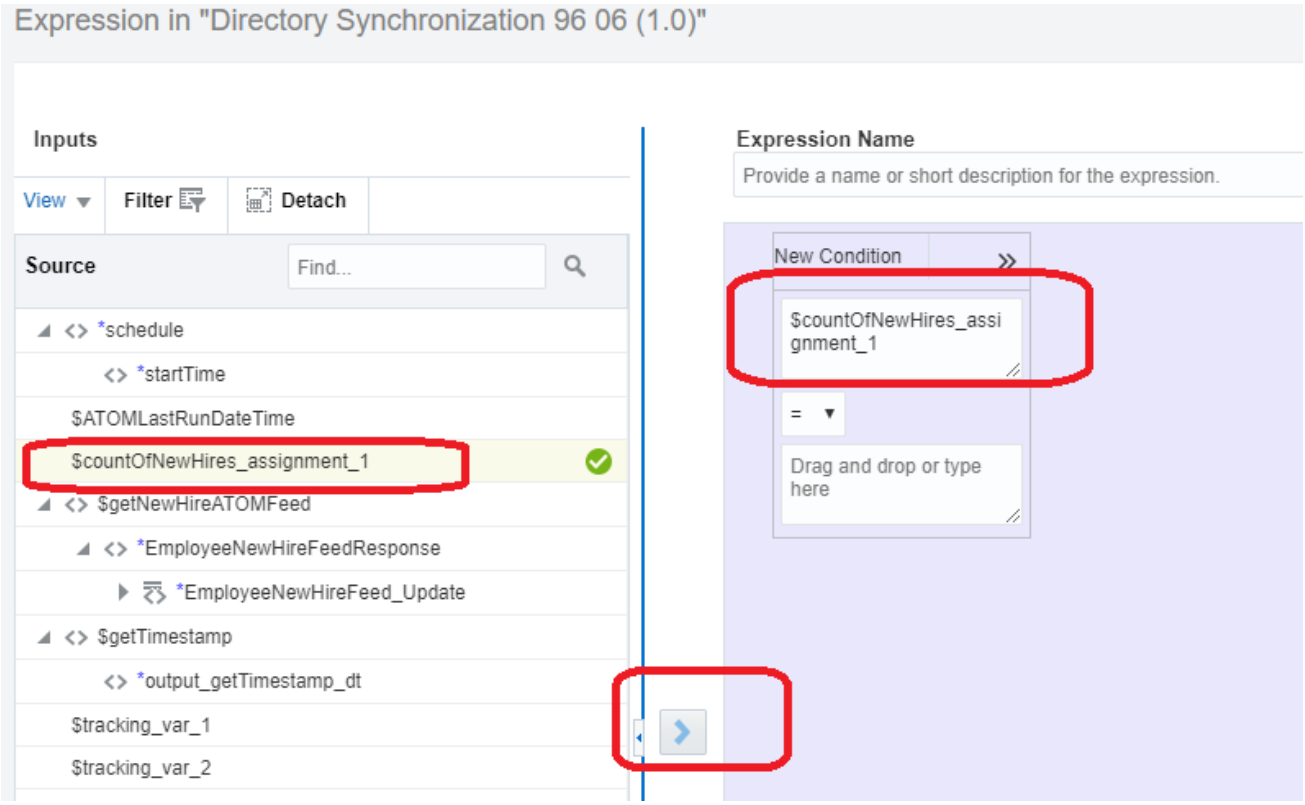
At this point in the integration it is necessary to identify whether there are any new records returned in the ATOM feed. This is achieved by using a **Switch a**ction.

1. On the right side of the integration canvas, click **Actions** to expand the panel.
2. Drag the **Switch** icon to the integration canvas to the **+** sign following the **countOfNewHires** label.

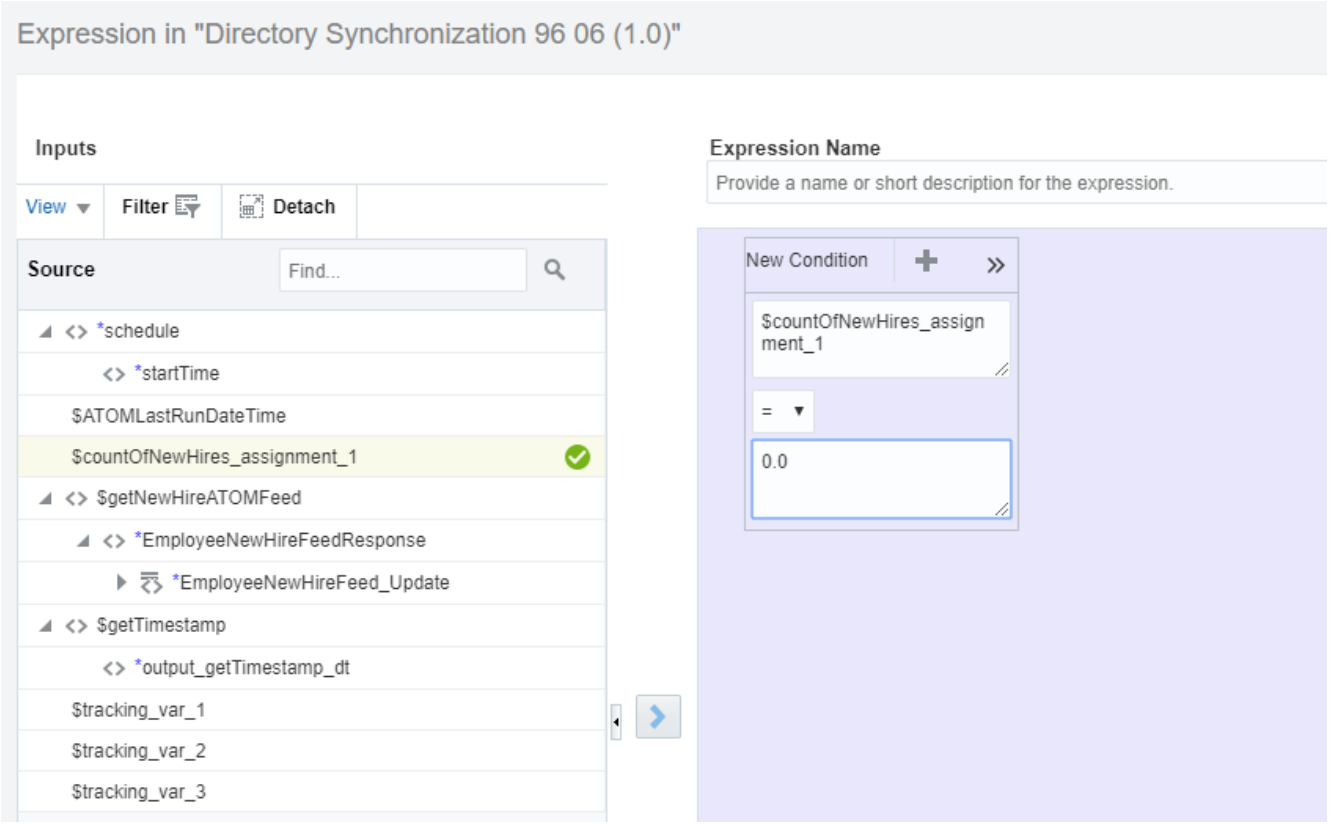
3. Two branches are automatically created:
   - **Undefined** (first) branch: This is the branch where the **countOfNewHires** will be validated for routing
   - **Otherwise** (second) branch: This branch is taken if the routing expression for the initial branch does not resolve to true.



4. Click the **Undefined** branch icon.
5. Select the **Edit** icon from the menu that is displayed. This invokes the **Expression** Builder.
6. From the **Source** component, select the **$countofNewHires_assignment_1** and click on > symbol to move it to the first text box for the new condition.



7. Select = as value in the drop down
8. Enter 0.0 is the second text box condition.
   This will result in the routing of the integration through this condition if there are no new hires returned within the ATOM feed.

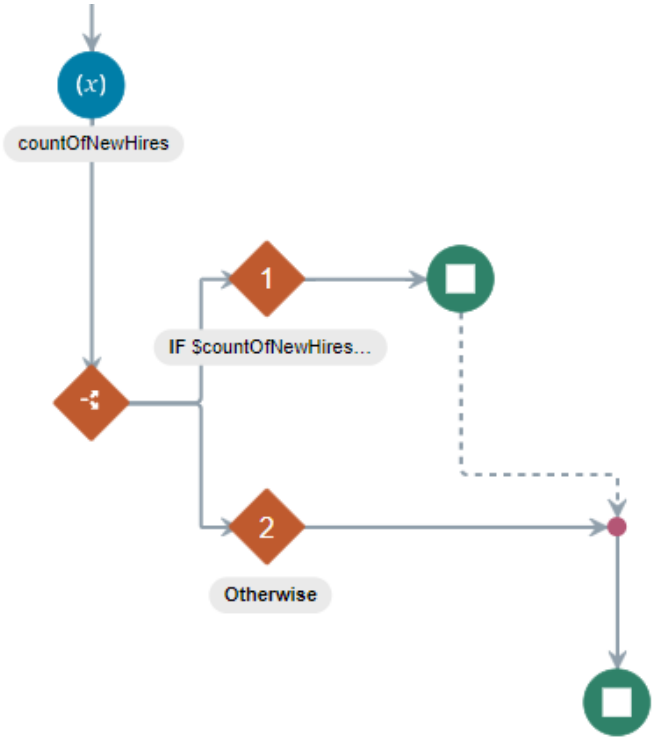


9. Click on Validate and Click Close. This displays the integration canvas
10. Click on Save

Add Stop Action

In order for the integration to complete when there are no new hires present in the ATOM Feed file, it is necessary to include a **Stop** action. This is achieved in the following way.

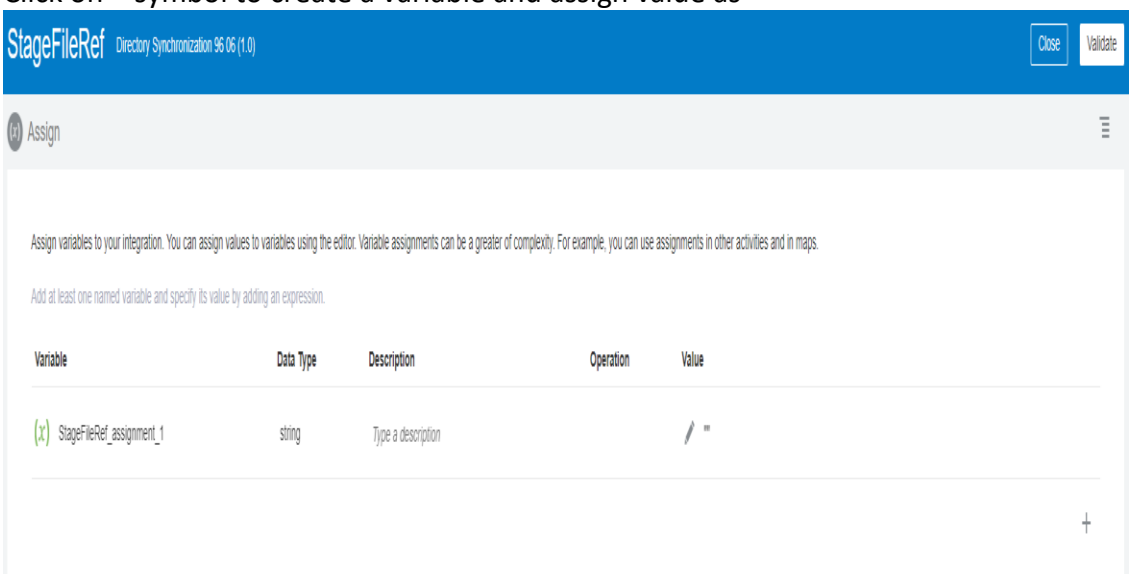
1. On the right side of the integration canvas, click **Actions** to expand the panel.
2. Drag the **Stop** icon to the integration canvas to the **+** sign following the **IF $countOfNewHires** label.
   This will add a new stop action to the canvas as shown below.



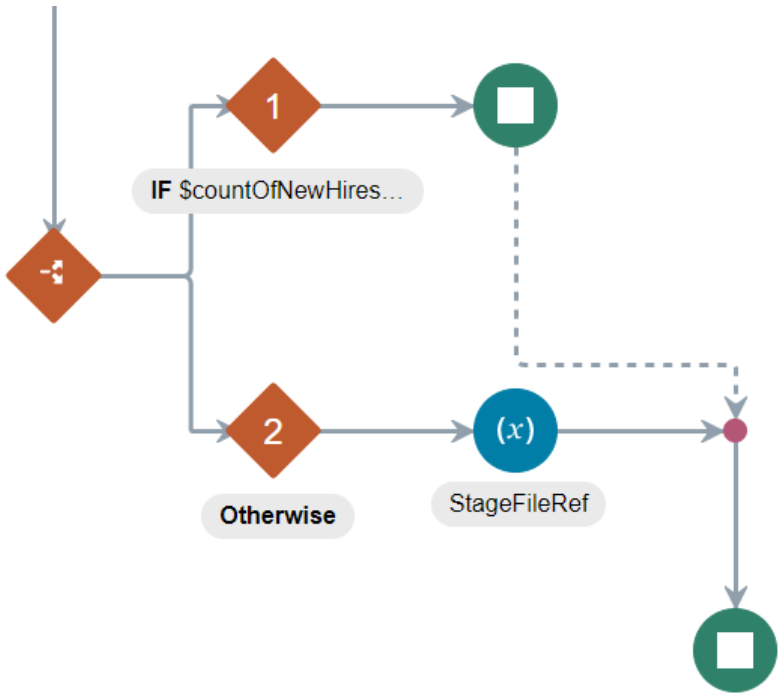
StageFileRef (Assign Activity)

The next task in the integration is to create a temporary variable to hold the file reference of a stage file activity, which we are going to perform in the next section.

1. Drag and drop Assign activity after Otherwise activity and enter the name as “StageFileRef”
2. Click on + symbol to create a variable and assign value as “”



3. Click on Validate and Close and Click on Save.

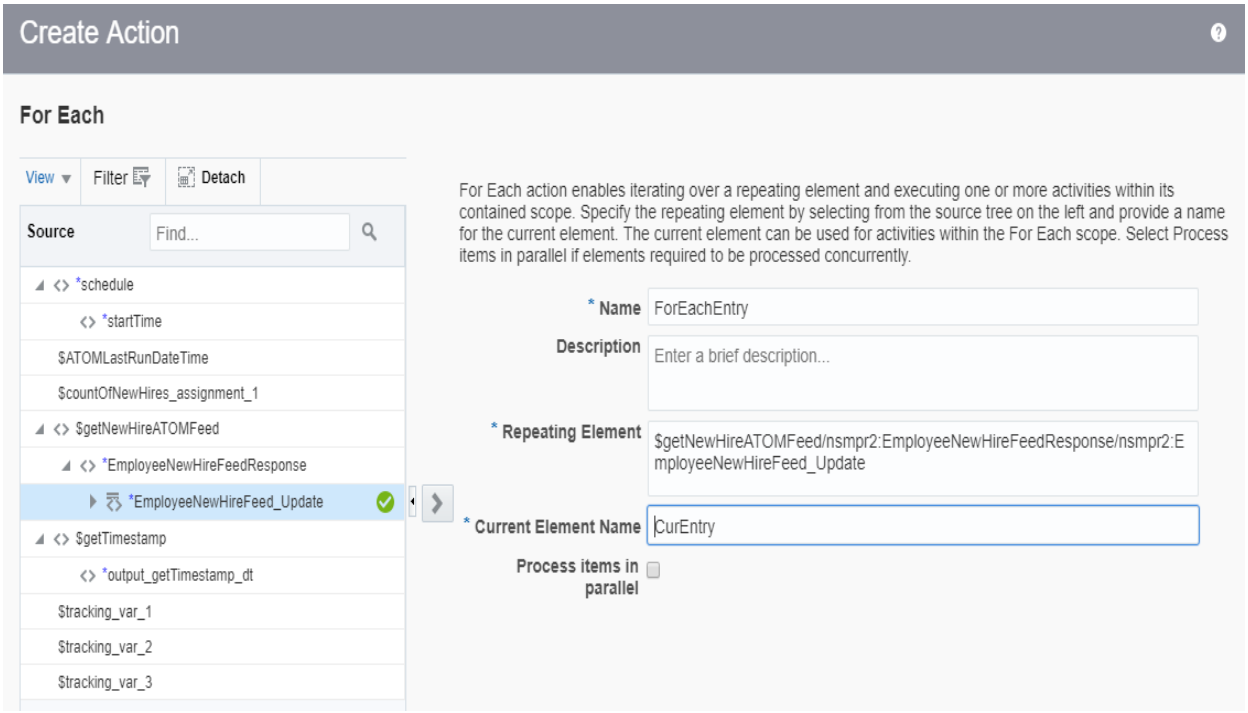


ForEachEntry
In order to retrieve the additional details required by the downstream Identity Management System, it is necessary to query HCM Cloud and build a richer data set than returned via the ATOM feed.

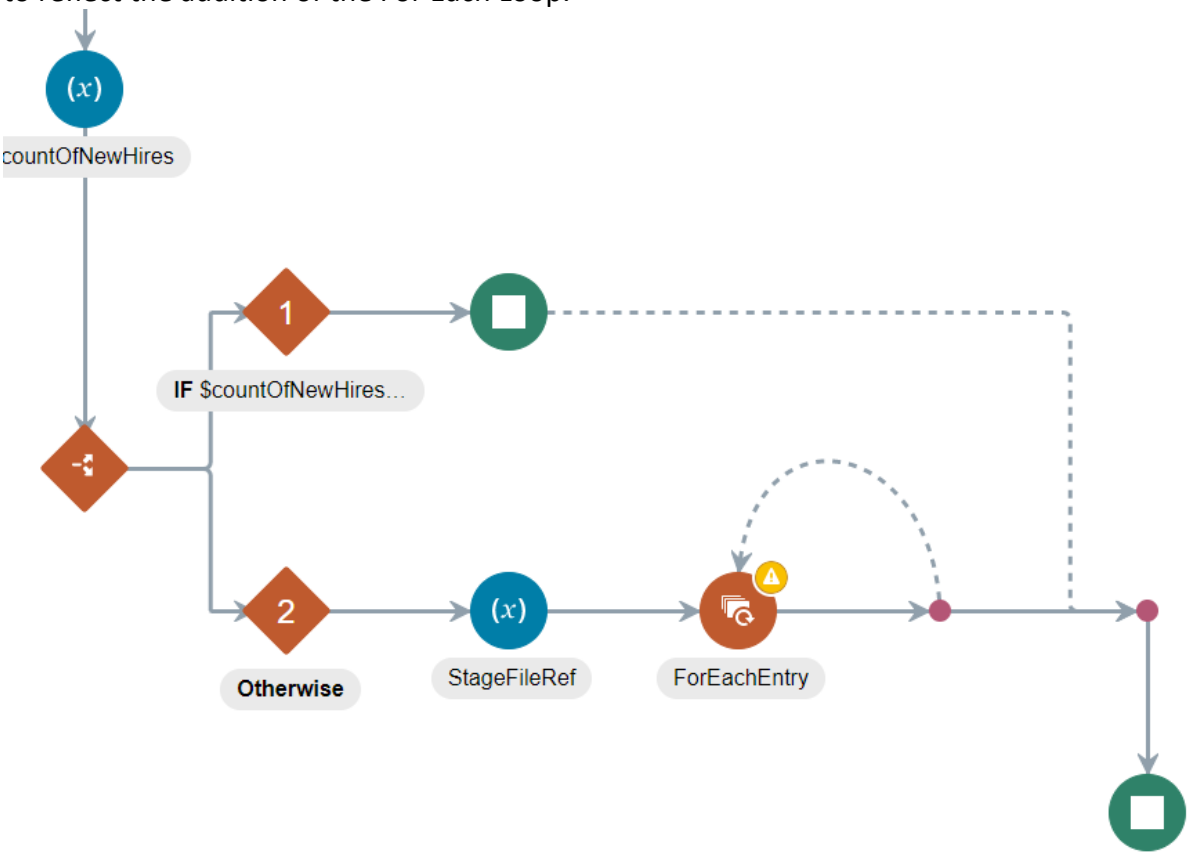
The **For Each** action enables you to loop over a repeating element and execute one or more actions within the scope of the **For Each** action. The number of loop iterations is based on a user-selected repeating element.

The most effective approach in retrieving this data is via the HCM Cloud REST service. The next action to be added to the integration canvas is a **ForEach Loop** to cycle through each individual new hire in the returned ATOM feed.

1. On the right side of the integration canvas, click **Actions** to expand the panel.
2. Drag the **For Each** icon to the integration canvas to the **+** sign following the **Otherwise** label. This will add a new **For Each** icon to the canvas and open the configure **Create Action** dialogue window.
3. Enter ForEachEntry as the Name of the action, refer the below screenshot in point number 5
4. Select the **EmployeeNewHireFeed_Update** element from the source component and click on > to move into the **Repeating Element,** refer the below screenshot in point number 5
5. Enter CurEntry as the **Current Element Name** value, refer the below screenshot
   The **EmployeeNewHireFeed_Update** is the parent level within the JSON ATOM Feed result file which represents an individual new hire. The **For Each** action will loop through the JSON file until each individual new hire has been processed.



6. Click **Create**. This will close the **Create Action** dialogue window updating the integration canvas to reflect the addition of the For Each Loop.

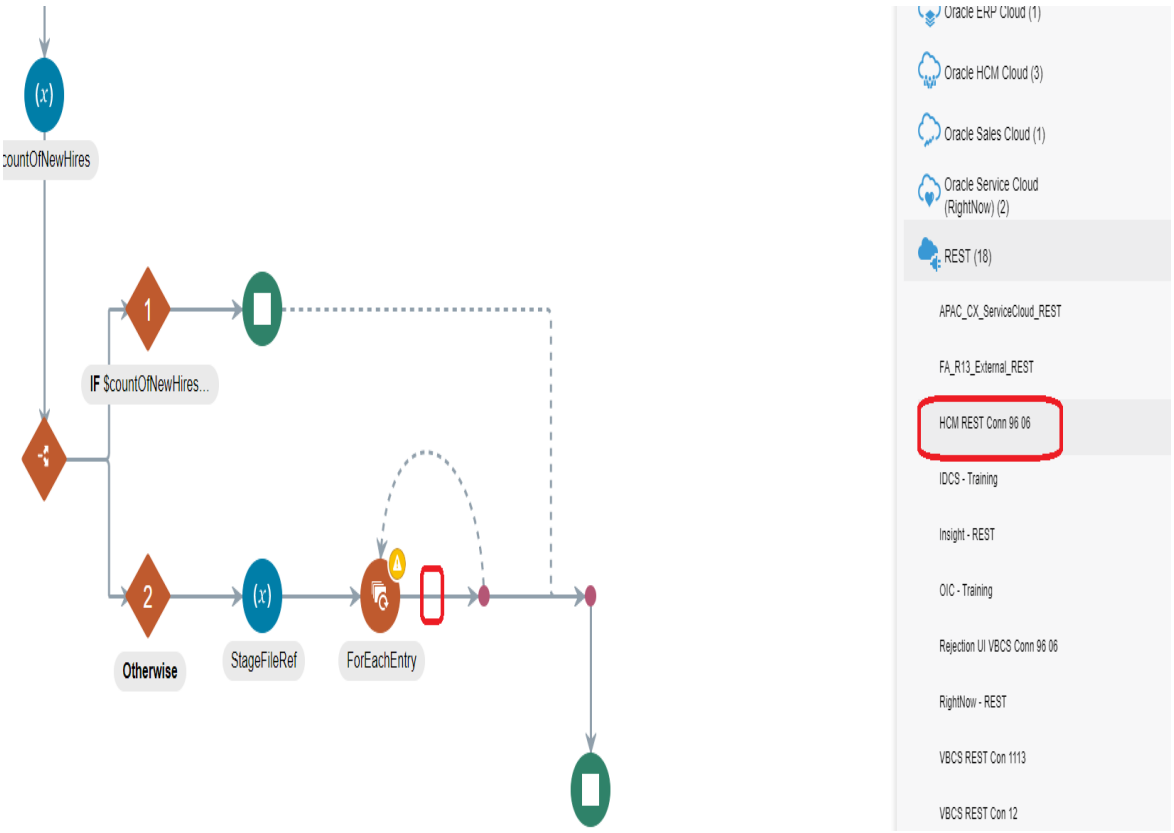


7. Click on Save

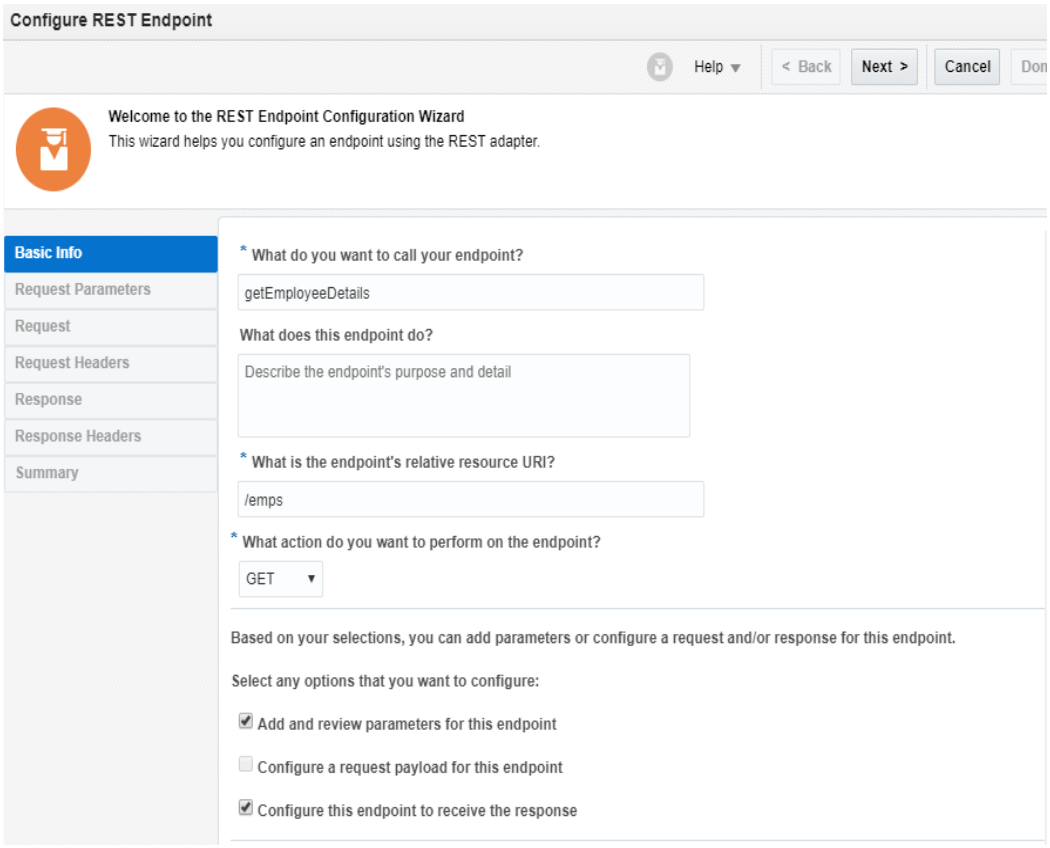
getEmployeeDetails (REST Adapter)

The next task in the integration is to make a REST call to HCM Cloud and retrieve the new hire details.

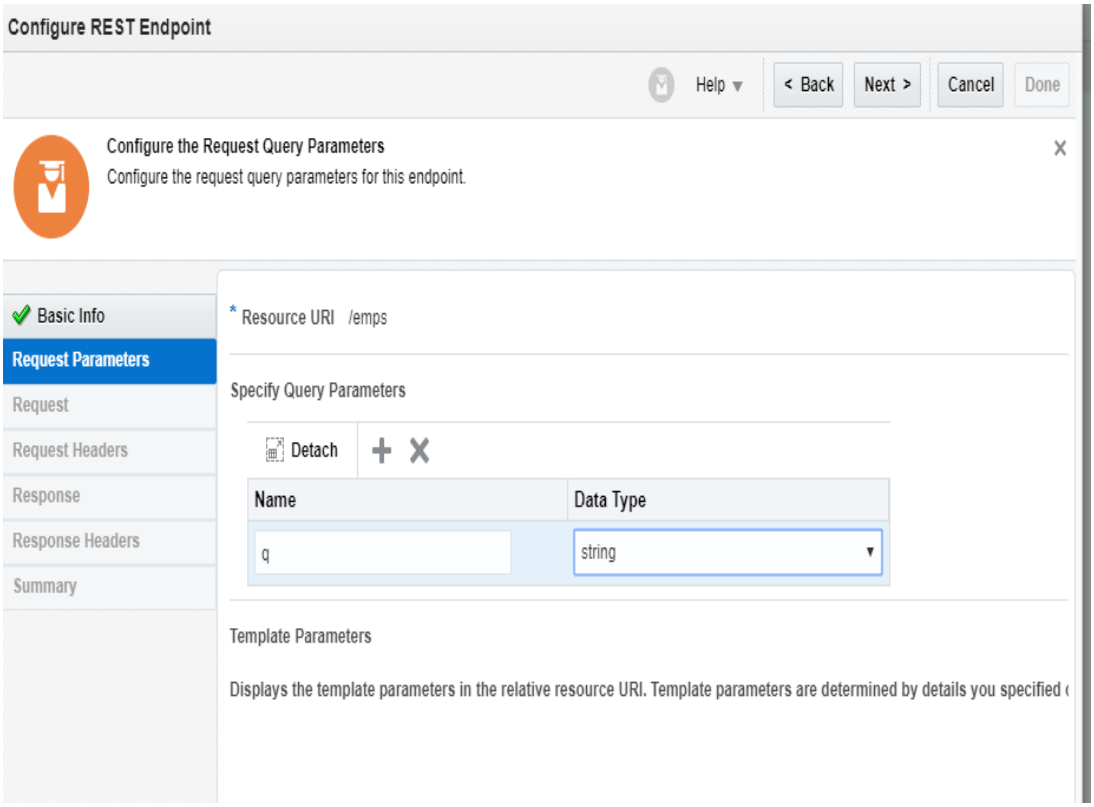
4. In the right navigation pane, click **Invokes**.
5. Click **REST**, then drag the HCM REST Conn 96 06 connection to the **+** sign following the **ForEachEntry** action and before the end of the arrow returning to the **ForEachEntry** Label. This ensures the call is repeated for each new hire.



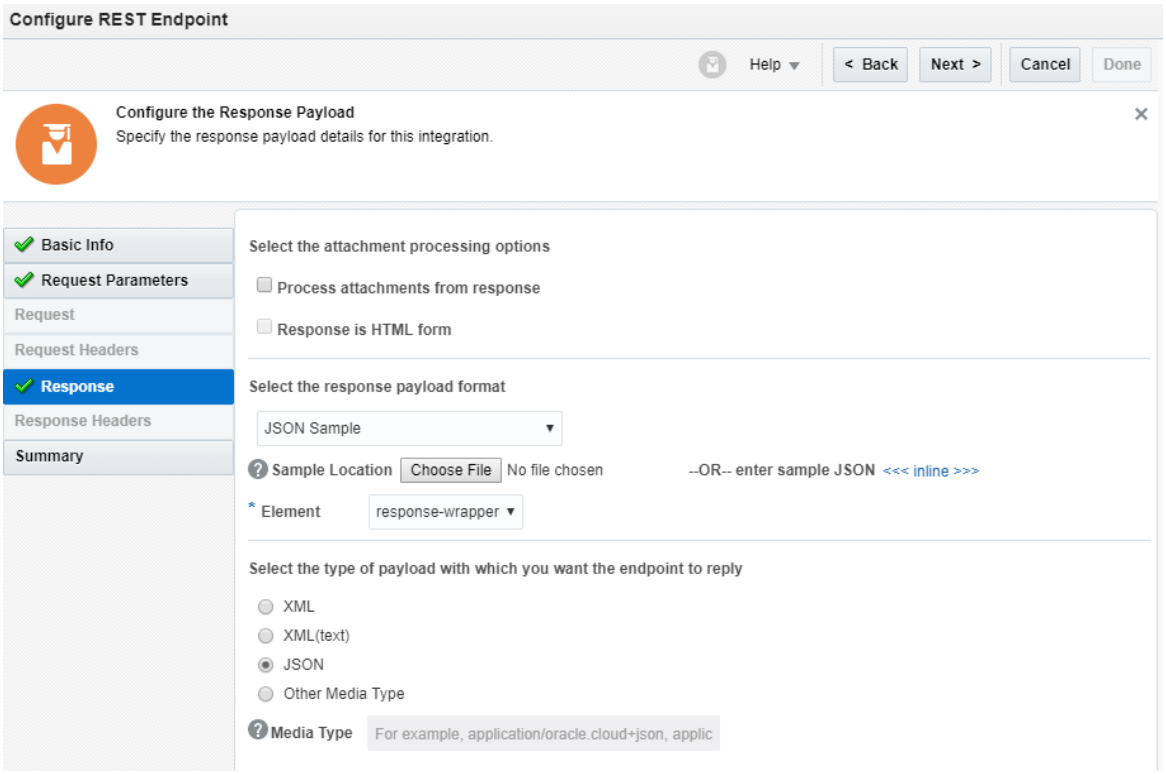
6. The **REST Adapter** configuration wizard is displayed.
7. Enter getEmployeeDetails in the **What do you want to call your endpoint?**
8. Enter /emps in the **What is the endpoint's relative resource URL?**
9. Select **GET** as the value for **What action does the endpoint perform?**
10. Check **Add and review parameters for this endpoint**
11. Check **Configure this endpoint to receive the response**



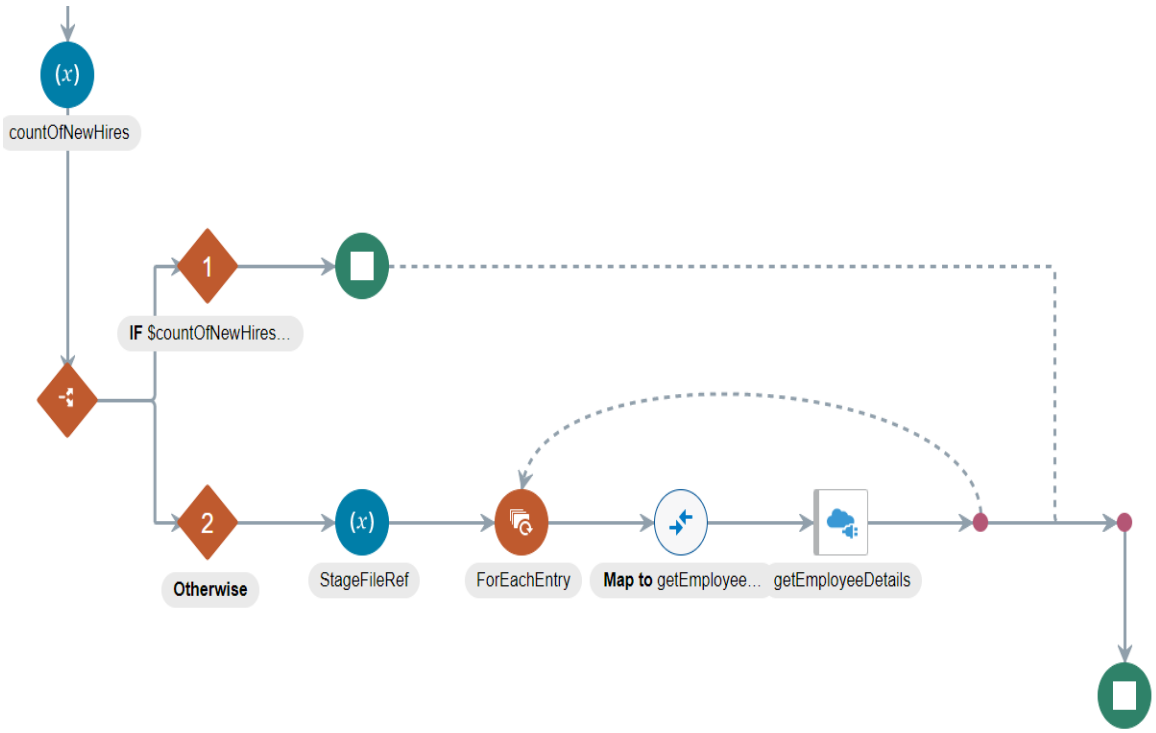
12. Click **Next**. This will display **Request Parameters** tab.
13. Enter q in the **Name** field. This represents the value that will be used as the input query for the REST call.
14. Select **string** as the value for **Data Type** drop down



15. Click **Next**. This will display **Response** tab.
16. Select **JSON Sample** as the value for **Select the response payload format** drop down
17. Click **<<<inline>>>** next to the **-OR- enter sample JSON label.** This will display **Response Sample Json Payload** tab.
18. Copy the contents of the **getEmployeeResponse.json** file into the text box. This file is part of the delivered artifacts with this guide.
19. Select **OK**. This will return the display to the **Response** tab.
20. Select the **JSON** radial value for the **Select the type of payload with which you want the endpoint to reply**.



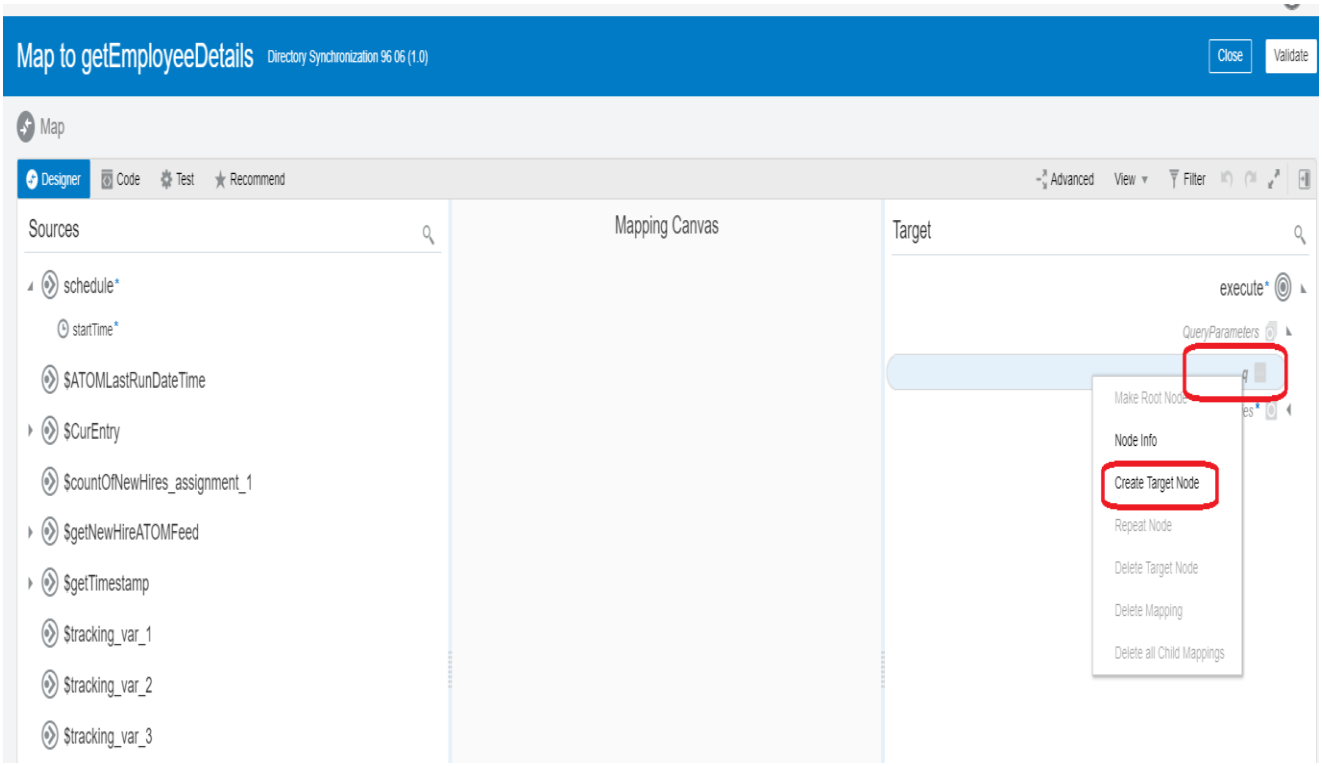
21. Click **Next**. This will display **Summary** tab.
22. Select **Done**. This will add both **getEmployeeDetails** REST Adapter and **Map to getEmployeeDetails** icons to the integration.



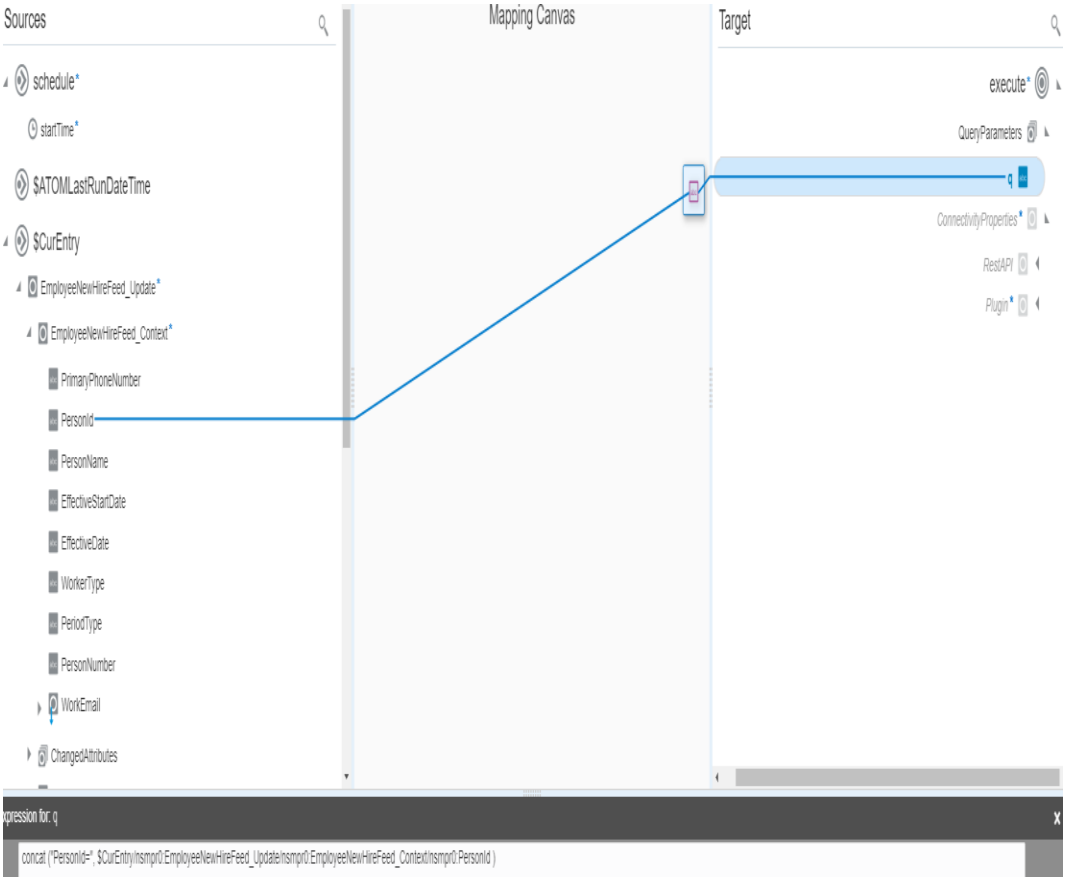
Map to getEmployeeDetails

Next it is necessary to define the inputs for the **getEmployeeDetails** step.

1. Click the Mapper icon (**Map to** getEmployeeDetails).
2. Click **Edit** to invoke the mapper.
3. Right click on **q** which is under execute→QueryParameters on the **Target** section. And click on Create Target Node. This will open the **Expression for: q** panel in the lower part of the UI.



4. Select from the **Toggle Functions**  icon to expose the available **Components** and **Operators**.
5. Select and drag the **string concat** field into the **Expression for : q** panel.
6. Enter "PersonId=" as the first line value
7. Select PersonId from CurEntry → EmployeeNewHireFeed_Update → EmployeeNewHireFeed_Context and drag into the second line value under the **concat** statement. At runtime this statement will concat the value of PersonId= with the actual value returned in the **PersonId** field for a new hire. This will construct the query value for the REST call made to HCM Cloud.
   The REST call being constructed will look like the following where [PersonIdValue] is the value returned from the ATOM feed for an individual Employee.
   https://<hostName>/hcmCoreApi/resources/latest/emps?q=PersonId=[PersonIdValue]

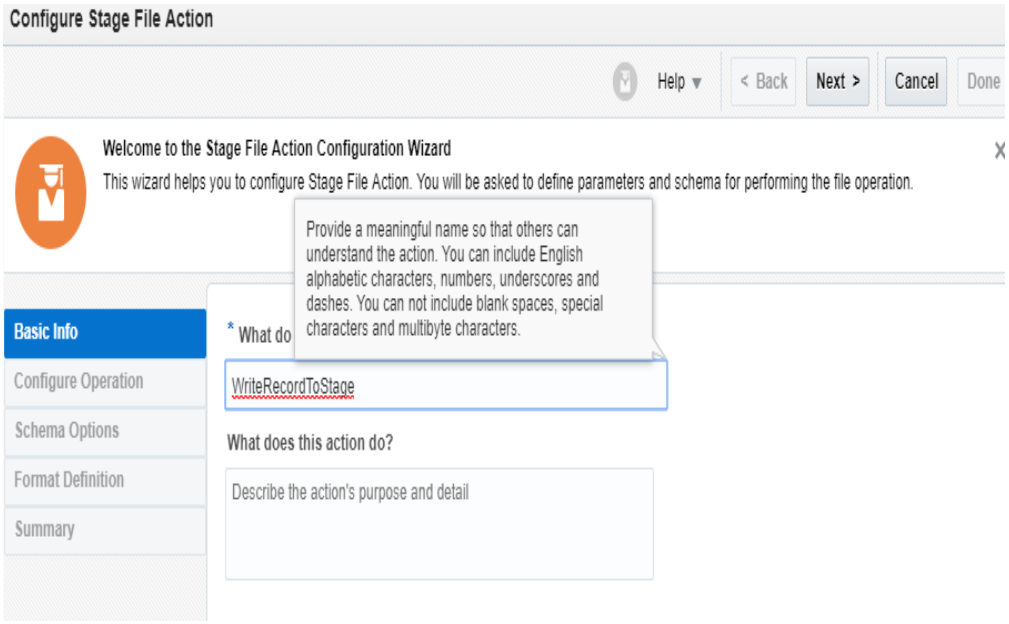


8. Select **Validate** and **Close**. This will display the integration showing a configured **Map to getEmployeeDetails** icon.
9. Click on Save to save Integration flow.

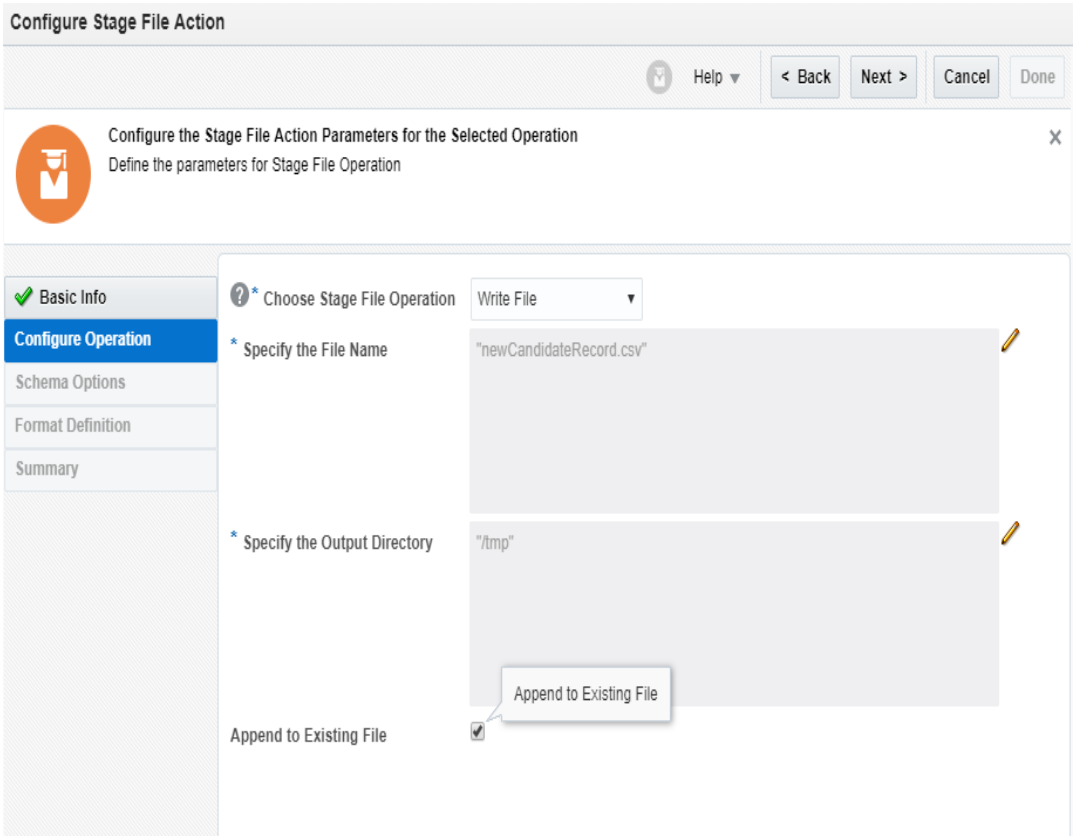
writeNewUserRecord to Staging

The next task in the integration is to write the new user record into the staging file using the Stage activity.

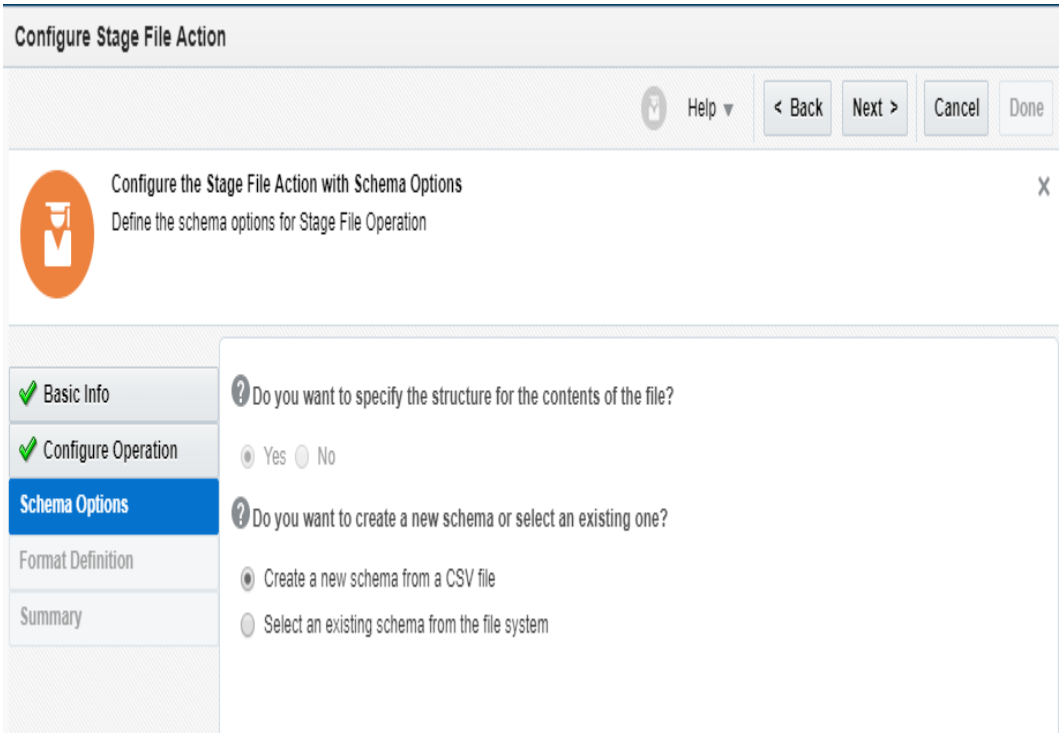
1. In the right navigation pane, click **Actions** and drag and drop Stage file activity after getEmployeeDetails.
2. Enter the “WriteRecordToStage” in **What do you want to call your action** and click on Next



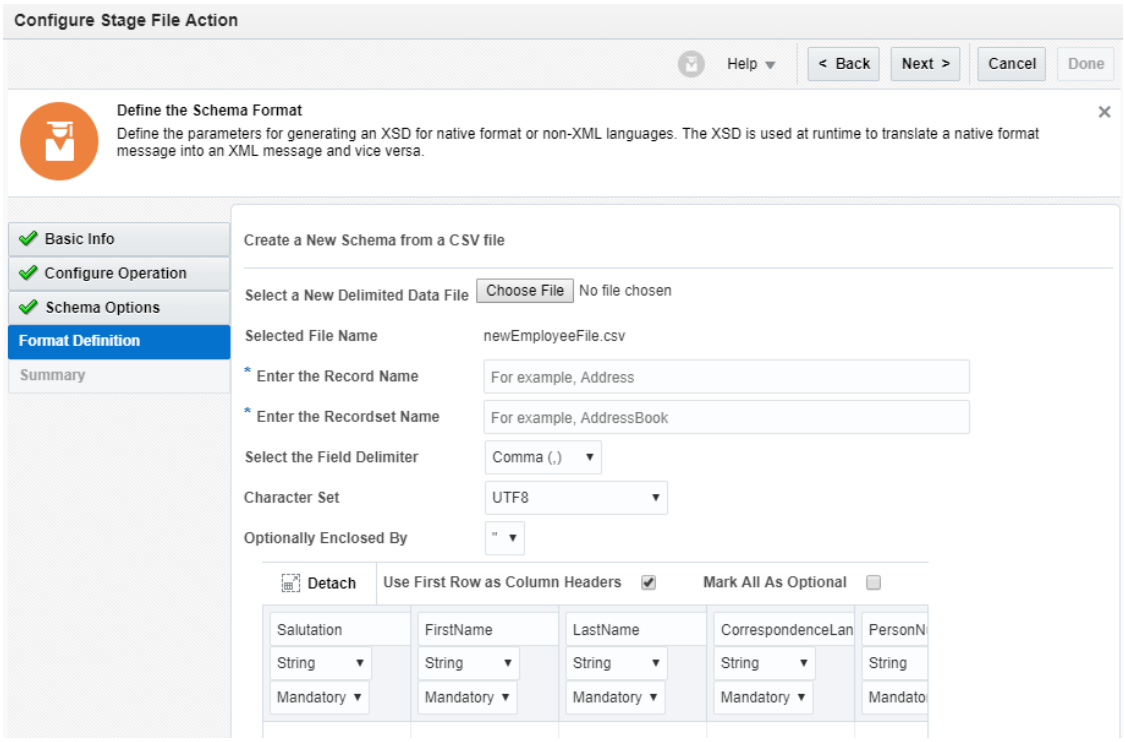
3. Choose Stage File Operation as “Write File” and Specify the file name as “newCandidateRecord.csv” and specify the Output Directory as “/tmp” and check the Append to Exist file option and click Next



4. Keep the default options as it is and click Next

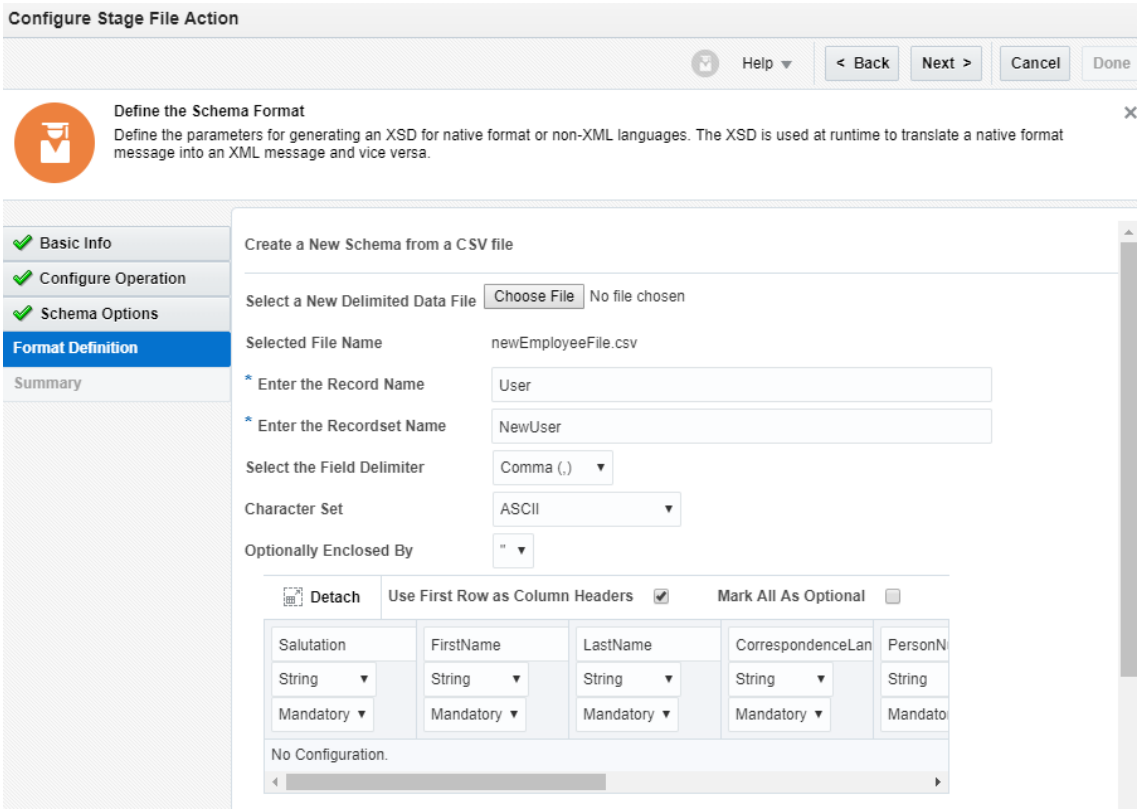


5. Click on “Choose File” button and select the file newEmployeeFile.csv which is given along with this document. The structure of the csv will be displayed in the lower half of the page.

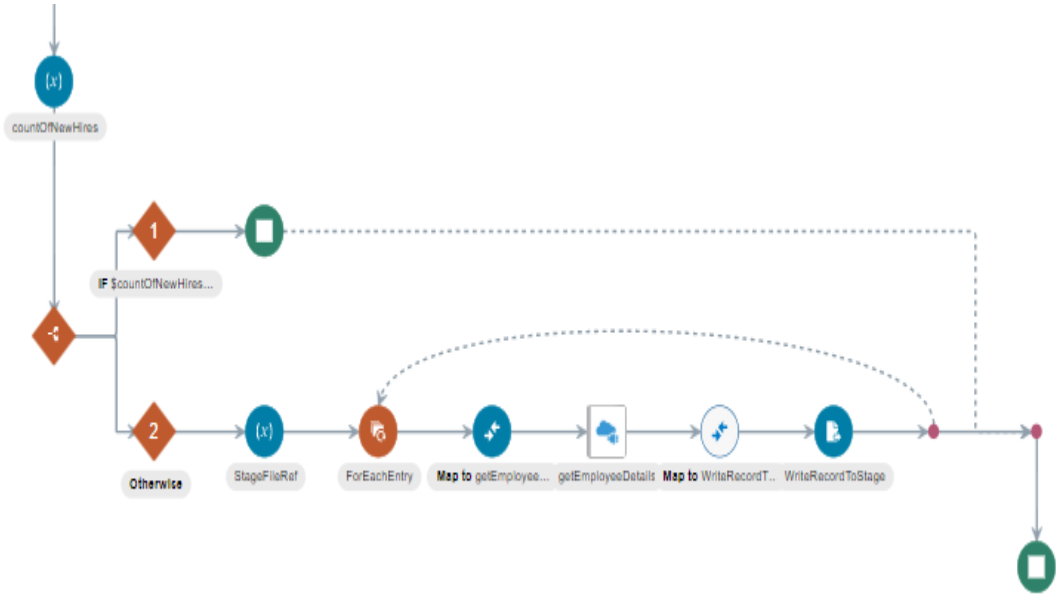


6. Enter User value for the Enter the **Record Name** field.
7. Enter NewUser for the Enter the **Recordset Name** field.
8. Select **Comma(,)** drop down value for **Select the Field Delimiter**
9. Select **ASCII** drop down value for **Character Set**

10. Select **“** drop down value for **Operationally Enclosed By**



11. Click **Next**. This will display **Summary** tab and click on Done. This will return you to the integration canvas.

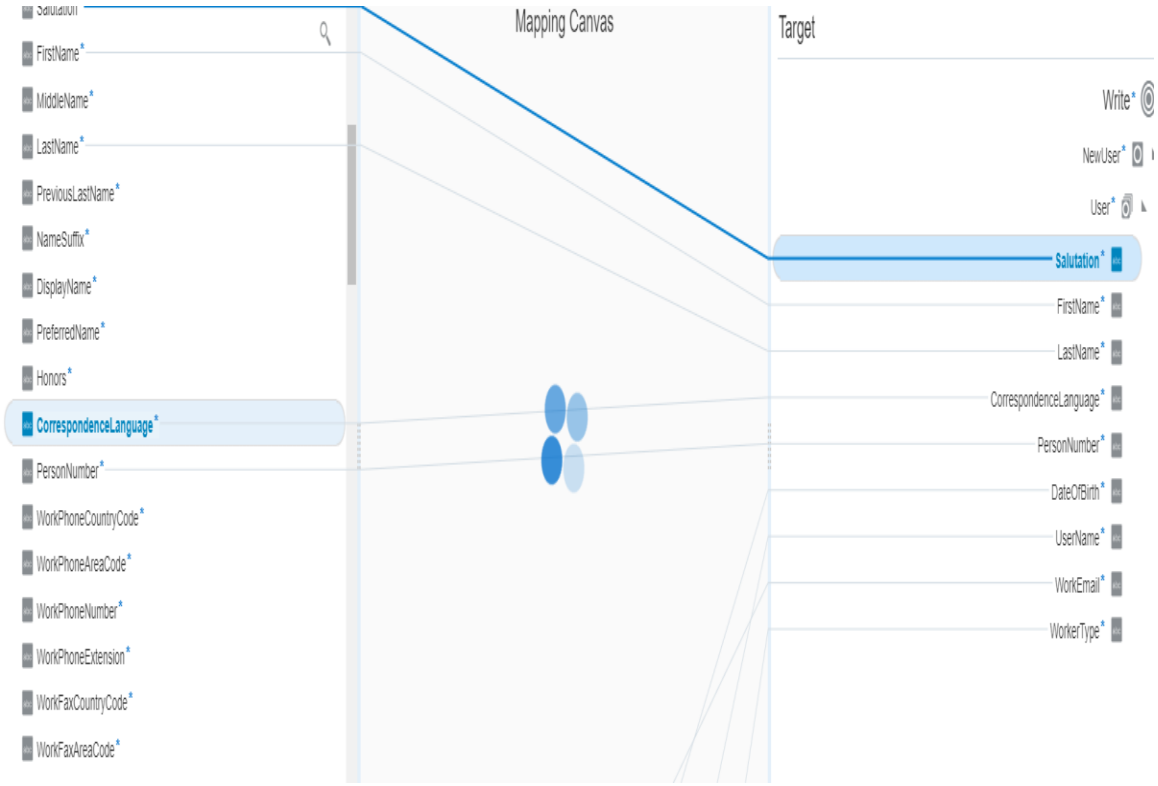


12. Click on Save

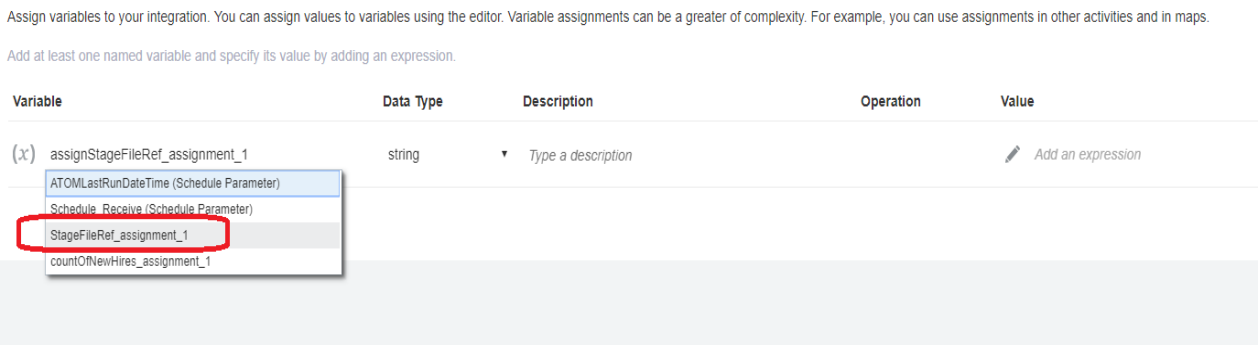
Map to WriteRecordToStage

This map to action will represent the transformation from JSON format to the CSV file format defined in the Stage action.

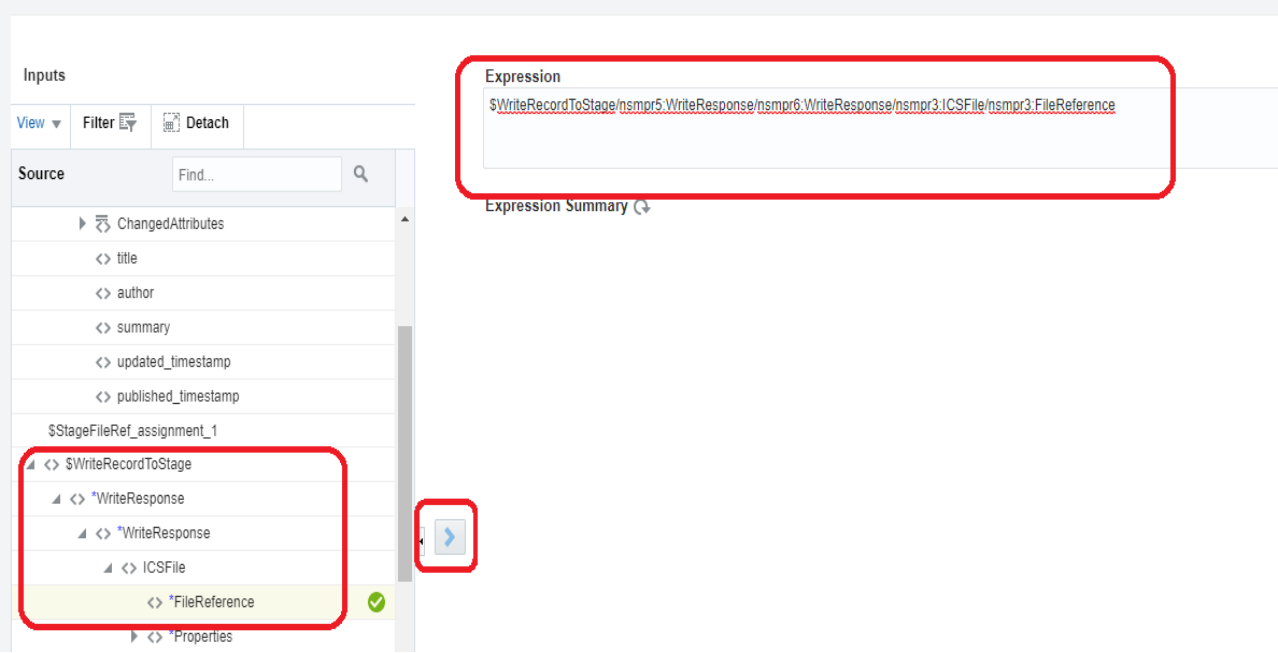
1. Click the Mapper icon (**Map to** WriteRecordToStage).
2. Click **Edit** to invoke the mapper.
3. Select the fields under the **getEmployeeDetails → executeResponse → response-wrapper → items** form the **Source** side of the UI and drag them to the corresponding **Target** values.
   The **Target** values are under **NewUser** and **User**. For the purposes of this guide select the fields which will generate a meaningful output file.



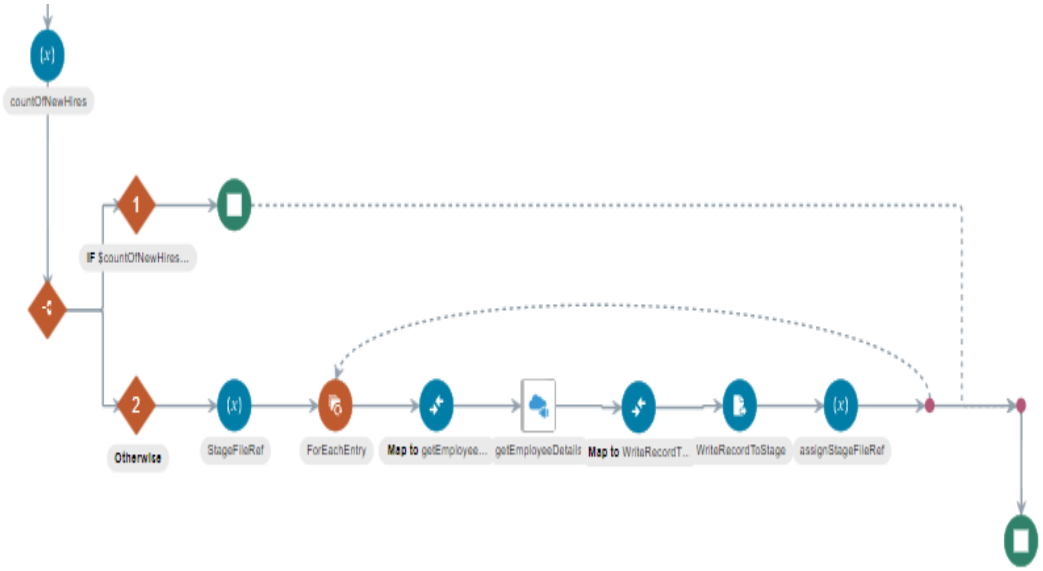
4. Click on Validate, Click on Close and Click on Save on Integration Canvas.
5. Drag and drop Assign activity after “writeRecordToStage” activity and enter name as assignStageFileRef and click on Create
6. Click on + symbol and select the Variable “StageFileRef_assignment_1” from drop down



7. Click on Expression Builder under Value
8. Expand WriteRecordToStage → WriteResponse→ WriteResponse → ICSFile and select FileReference and click on > symbol



9. Click on Validate and close. Click on Validate and close and Click on Save.



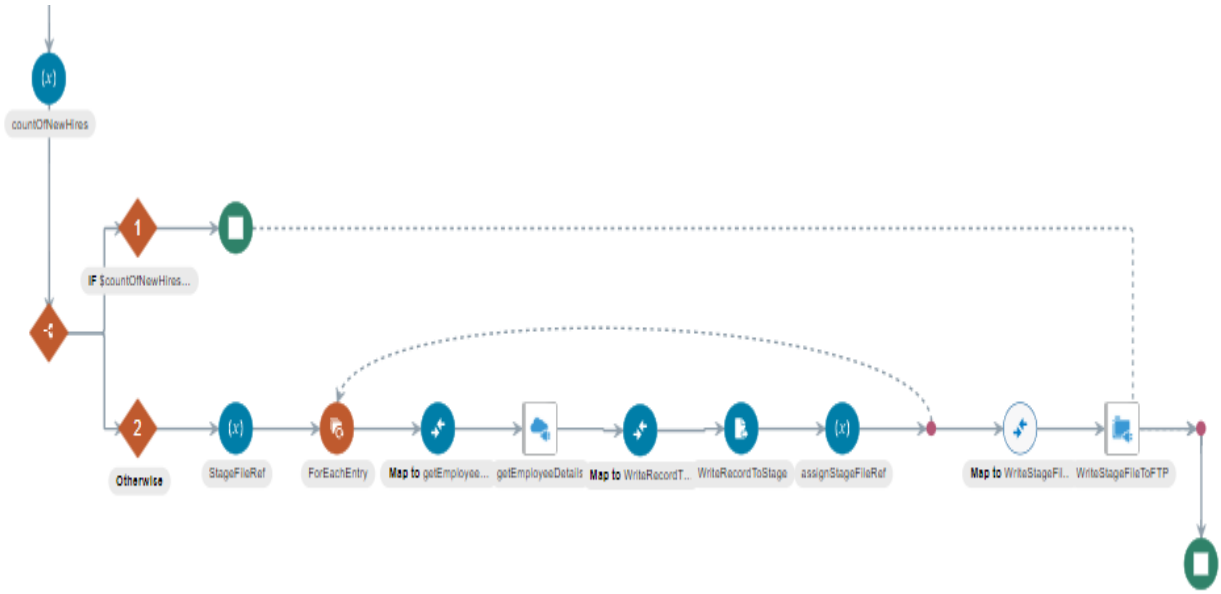
WriteStageFileToFTP (FTP Adapter)

The next task in the integration is to add the **FTP Adapter** to the integration representing the step where the Stage file is written to the customer FTP server.

13. In the right navigation pane, click **Invokes**.
14. Click **FTP**, then drag the "FTP Conn 96 06" connection to the **+** sign after **ForEachEntry** loop and at the end of the Otherwise section. The **FTP Adapter** configuration wizard is displayed.
15. Enter WriteStageFileToFTP in the **What do you want to call your endpoint?**
16. Click **Next**. This will display **Operations** tab.
17. Select the **Write File** value in the **Select Operation** drop down.
18. Select the **ASCII** radial selection for the **Select a Transfer Mode**
19. Enter the **Output Directory** value location based on your FTP folder structure configuration given below. Please create the directory structure if it is not there in FTP location.
    `/upload/public_ftp/<<YOURNAMES>>/directsynch`

20. Enter the **File Name Pattern** value. Depending on the specific requirements, the file name can follow different pattern structures. A simple example would be as follows:

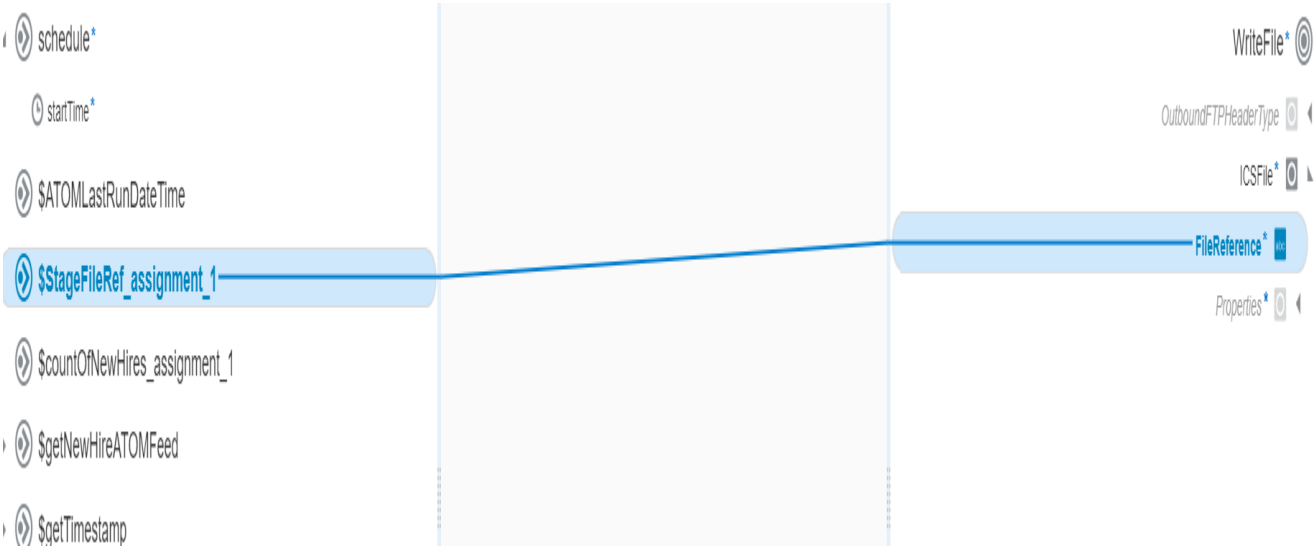
    newCandidateRecord%yyyyMMddHHmmss%.csv which would result in a file named newCandidateRecord20180418082100.csv for a file written on the 18th of April 2018 at 08:21:00 AM.
21. Click **Next**. This will display **Schema** tab.
22. Select the **No** value for the **Do you want to specify the structure for the contents of the file?**
23. Click **Next**.
24. Click **Done.** returning to the integration canvas. This will add both **WriteStageFileToFTP** and a **Map to WriteStageFileToFTP** icon to the integration.



Map to WriteStageFileToFTP

This map to action will map the file reference to FTP adapter.

10. Click the Mapper icon (**Map to WriteStageFileToFTP**).
11. Click **Edit** to invoke the mapper.
12. Select the StageFileRef_assignment_1 on the **Source** side of the UI and drag it to the FileReference under ICSFile on **Target**.



13. Click Validate and Click on **Close.** This will return you to the integration canvas. And click on Save

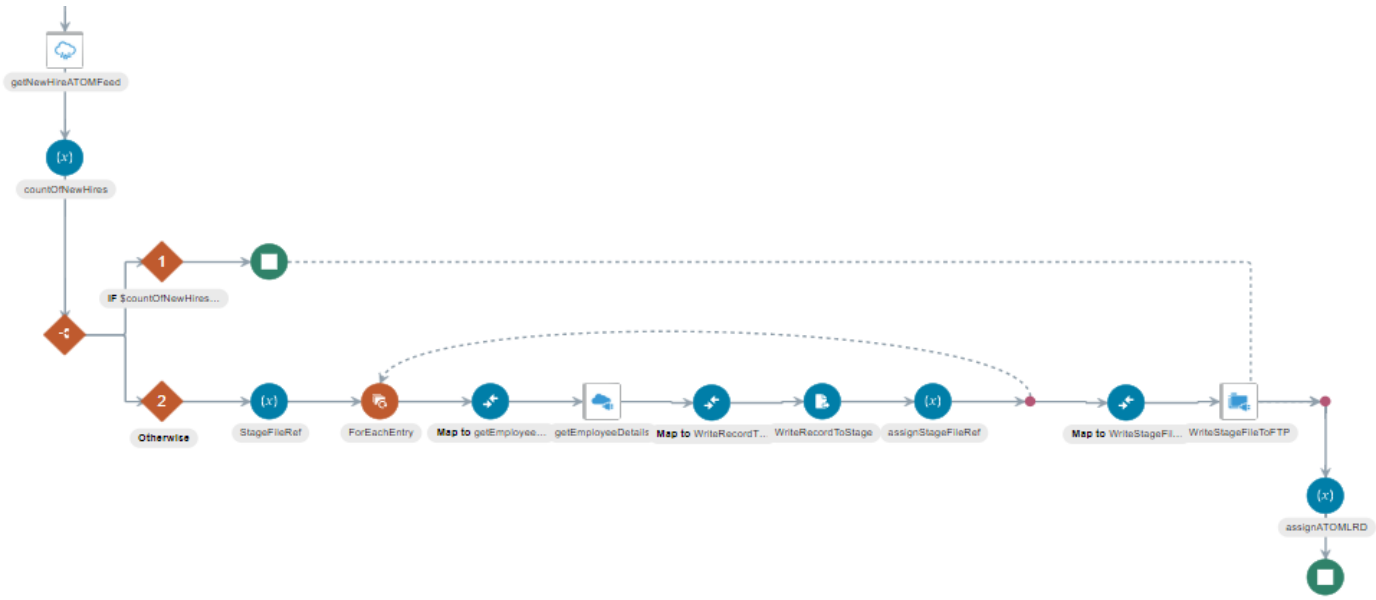
assignATOMLRDT

The final step in defining the integration is to set the ATOM Last Run Date for use next time the integration is run.

1. In the right navigation pane, click **Action**

2. Drag the **Assign** icon to the **+** sign following the WriteStageFileToFTP (after Switch condition) and before the **Stop** icon.
3. Enter assignATOMLRD as the **Name** value for the newly created action.
4. Select **Create**. This will open the **Assign** configuration page. Click on + symbol

5. Select the **ATOMLastRunDateTime (Schedule Parameter)** in the **Variable** drop down.
6. Select the  pencil icon in the **Value** field. This will open **Expression** dialogue window.
7. Select and drag and the **OutputGetTimestamp_dt** into the **Expression** field.
8. Click on Validate and Click on **Close**. The **Assign** configuration page will be displayed.
9. Click on Validate and Click on **Close**. This displays the integration canvas.
10. Click on Save



Tracking

The final step in configuring the integration is the setting of the business identifier. This is used to provide a unique value for monitoring and reporting.

1. From the action menu on the integration canvas, select the **Tracking** option. This will open the **Tracking** dialogue window
2. Drag the **startTime** value from the Source panel on to the first **Tracking Field**.
3. Select **Save**. This is will close the **Business Identifiers for Tracking dialogue** window and return you to the integration canvas.
4. Select **Save**. You are now ready to activate and execute the integration.

Activate and Run the Integration.

1. Find your integration flow on the list and click the **Activation switch**. This will open the **Activate Integration** dialogue window.
2. Check the **Enable tracing**
3. Check the **Include payload**
4. Select the **Activate** button
5. Once the **Activation** is complete, select from the action menu the **Submit Now** action.
6. This will execute the integration. A **request id** value will be displayed.
7. Click the **request id** value. This will open the **Monitor Runs** page. Here you can view the execution of the integration. Once the integration has completed a Successful notification will be displayed.
8. Select the **RUN ID** value. This will open the Tracking summary page for this integration.
9. Select the **start Time:** value (which is the **Business Identifier** set on the integration earlier). This will open the integration instance screen, displaying all of the steps within the integration. They will be green, indicating that they were successful.

Validating Extracted File presence on FTP Server

Following the execution of the integration within OIC, you should navigate to the folder on your FTP server to validate that the file is located there.